INSTRUCTION MANUAL

> DataVS2

DATALOGIC

Datalogic S.r.l.
Via S. Vitalino 13
40012 Calderara di Reno
Bologna - Italy

DataVS2 Instruction Manual

Ed.: 12/2017

© 2011 – 2017 Datalogic S.p.A. and/or its affiliates ◆ ALL RIGHTS RESERVED. ◆ Without limiting the rights under copyright, no part of this documentation may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means, or for any purpose, without the express written permission of Datalogic S.p.A. and/or its affiliates.

Datalogic and the Datalogic logo are registered trademarks of Datalogic S.p.A. in many countries, including the U.S.A. and the E.U.

Datalogic shall not be liable for technical or editorial errors or omissions contained herein, nor for incidental or consequential damages resulting from the use of this material.

22/12/17

CONTENTS

COMPLIANCE vii
CE Compliance vii
FCC Compliance vii

GENERAL VIEW viii
LED Description viii

1 GENERAL INFORMATION 1
1.1 General Description 1
1.2 Components 1

2 INSTALLATION 2
2.1 Mounting 2
2.2 Operating Distances 3
2.3 Focusing Ring 4
2.4 Electrical Connections 4
2.4.1 X1 Connector (M12 8-pin): Power Supply and I/O 5
2.4.2 X2 Connector (M12 4-pin): Ethernet Connectivity 5
2.5 Graphic User Interface installation and Start-Up 6
2.5.1 Minimum System Requirements 6
2.5.2 DataVS2 Graphic User Interface Installation 7

3 SENSOR SETUP 8
3.1 General Overview 8
3.2 Communicating with the Sensor 9
3.2.1 Check PC Network Settings 10
3.2.2 Set up Communication 12
3.2.3 Set up Sensor IP-Address 13
3.2.4 Troubleshooting 15
3.3 Updating Software 15
3.4 General Sensor Settings 16
3.4.1 Network Settings 18
3.4.2 Memory Status 19
3.4.3 Firmware Update 20
3.4.4 Firmware Update Rules 22
3.4.5 General Settings 23
3.4.6 Serial Communication and Ethernet Parameters 24

4 INSPECTION SETUP 26
4.1 General Considerations 26
4.2 STEP 1: IMAGE SETUP 27
4.2.1 Quick Guide STEP 1 27
4.2.2 Operating Modes 28
4.2.3 Establish Connection with Sensor – Online Mode Only 28
4.2.4 Task Selection 30
4.2.5 Basic Settings – Online Mode Only 32
4.2.6 Trigger Selection 33
4.2.7 Reference Image Setup 36
4.3 STEP 2: TEACH 36
4.3.1 Quick Guide STEP 2 36
4.3.2 Locator Selection 38
4.3.3 Control Selection 39

4.3.4 Configuration of Outputs and Image Saving Condition 41
4.3.5 Example of a Timing Diagram 45
4.3.6 Logical Tool Selection 47
4.4 STEP 3: RUN 49
4.4.1 Quick Guide STEP 3 49
4.4.2 Run Settings 51
4.4.3 Test 52
4.4.4 Run 53
4.5 Inspection Settings 54

5 DATAVS2 EASY GRAPHIC USER INTERFACE 56
5.1 DataVS2 GUI Components 56
5.1.1 Menu Bar 57
5.1.2 Tool Bar 61
5.1.3 Setup Panel 64
5.1.4 Control Panel 64
5.1.5 Image Buffer 65
5.1.6 Work Area Tabs 66
5.1.7 Inspection Explorer 70
5.1.8 Help 71
5.1.9 Status Bar 71
5.2 Image Processing Tools 72
5.2.1 Locators and Controls 73
5.2.2 DataVS2 Tools 74
5.3 Locators 75
5.3.1 Usage of Locators 76
5.3.2 Selection and Positioning 77
5.3.3 Position Locator 78
5.3.4 Pattern Match Locator 81
5.3.5 360° Pattern Match Locator 85
5.3.6 360° Contour Match Locator 90
5.3.7 Datamatrix Locator 98
5.3.8 Barcode Locator 102
5.3.9 QRCode Locator 106
5.4 Controls 111
5.4.1 Selection and Positioning 111
5.4.2 Brightness 113
5.4.3 Contrast 115
5.4.4 Contour Match 119
5.4.5 Edge Count 122
5.4.6 Width 125
5.4.7 Pattern Match 128
5.4.8 Position 133
5.4.9 Datamatrix 137
5.4.10 Barcode 143
5.4.11 QRCode 148
5.4.12 OCV 154
5.4.13 360° Contour Counter 159
5.4.14 360° Defect Finder 166
5.5 Logical Tools 175
5.5.1 Logical Tool Selection and Configuration 176
5.5.2 Configuration of the Image Saving Outputs 179
5.5.3 Logical AND 180
5.5.4 Logical OR 180
5.5.5 Logical NOR 180

5.5.6 Logical XOR 181
5.5.7 Logical NAND 181
5.5.8 Logical XNOR 181
5.5.9 Logical NOT 182
5.5.10 Logical MAJORITY 182

6 INSPECTION SWITCHING 184
6.1 Inspection Switching Through DataVS2 GUI 184
6.2 Inspection Selection Through Digital Input 185
6.2.1 Standard Protocol 185
6.2.2 Expert Protocol 187
6.3 Inspection Switching Through Ethernet Communication 191
6.4 Inspection Switching Though RS232 Serial Communication Interface 192

7 TEACH IN 193
7.1 Mechanical Teach In 193
7.2 Digital INPUT Teach In 194
7.3 Teach In SOFTWARE 196
7.4 Teach In with vsm 198

8 READING INSPECTION RESULTS 199
8.1 Inspection Monitoring Through the DataVS2 GUI 199
8.2 Reading Inspection Results Through Custom Ethernet Communication 199
8.3 Reading Inspection Results Through RS232 Serial Interface 201

9 COMMUNICATION PROTOCOLS 202
9.1 Ethernet Communication Configuration 203
9.2 RS232 Serial Communication Configuration 204
9.3 Communication Protocol Grammar 205
9.3.1 Command Message Format 205
9.3.2 Response Message Format 206
9.3.3 Inspection Results Message Format 207
9.3.4 Inspection Results Message Configuration Through DataVS2 GUI 216
9.4 Commands Description 216

10 RECOVERY MODE 220
10.1 Recovery Procedure 221
10.1.1 Recovery Procedure for DataVS2 Basic Sensor 221
10.1.2 Recovery Procedure for DataVS2 Advanced Sensor 221
10.1.3 Recovery Procedure for DataVS2 Identification Sensor 222
10.1.4 Recovery Procedure for DataVS2 Professional Sensor 223

11 INSPECTIONS AND PERIODIC MAINTENANCE 224
11.1 Warranty 224

12 SPECIFICATIONS 225
12.1 DataVS2 Sensor Specification 225
12.2 Imager Sensor Specification 225

13 LIST OF AVAILABLE MODELS 227

14 OVERALL DIMENSIONS 228

15 ACCESSORIES 229

GLOSSARY.....231

COMPLIANCE

For installation, use and maintenance it is not necessary to open the reader.

Only connect Ethernet and dataport connections to a network which has routing only within the plant or building and no routing outside the plant or building.

CE COMPLIANCE

CE marking states the compliance of the product with essential requirements listed in the applicable European directive. Since the directives and applicable standards are subject to continuous updates, and since Datalogic promptly adopts these updates, therefore the EU declaration of conformity is a living document. The EU declaration of conformity is available for competent authorities and customers through Datalogic commercial reference contacts. Since April 20th, 2016 the main European directives applicable to Datalogic products require inclusion of an adequate analysis and assessment of the risk(s). This evaluation was carried out in relation to the applicable points of the standards listed in the Declaration of Conformity. Datalogic products are mainly designed for integration purposes into more complex systems. For this reason it is under the responsibility of the system integrator to do a new risk assessment regarding the final installation.

Warning: This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

FCC COMPLIANCE

Modifications or changes to this equipment without the expressed written approval of Datalogic could void the authority to use the equipment.

This device complies with PART 15 of the FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference which may cause undesired operation.

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

GENERAL VIEW

DataVS2

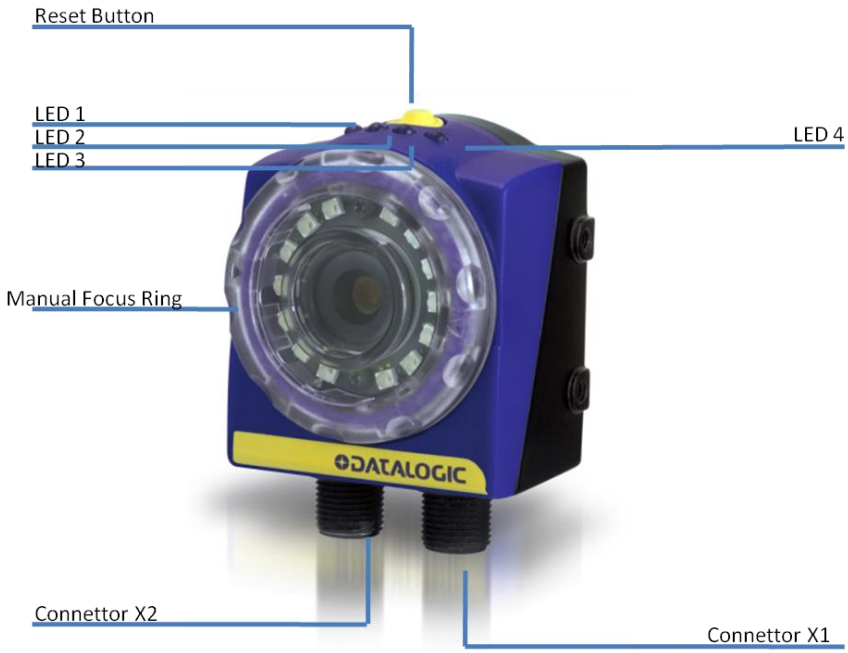


Figure A

LED DESCRIPTION

The following system operation LEDs are available on the sensor housing:

LED Number	Description	Working Colour
LED 1	Power connection	Green
LED 2	Digital output 1	Orange
LED 3	Digital output 2	Orange
LED 4	Network connection	Green

1 GENERAL INFORMATION

1.1 GENERAL DESCRIPTION

The DataVS2 of Datalogic S.p.A. is an easy-to-use vision sensor offering advanced inspection features that make it suitable for a broad range of applications (such as quality control on manufacturing lines or automated material handling, just to name a few).

Basically, it compares a digital image acquired at a certain instant (the "target image") with a stored reference image (the "master image").

Once the sensor is connected to a Personal Computer and configured using the supplied software, it offers stand-alone (i.e. fully independent) operation. Digital outputs only operate in the stand-alone mode and are disabled during the configuration process or in off-line operation.

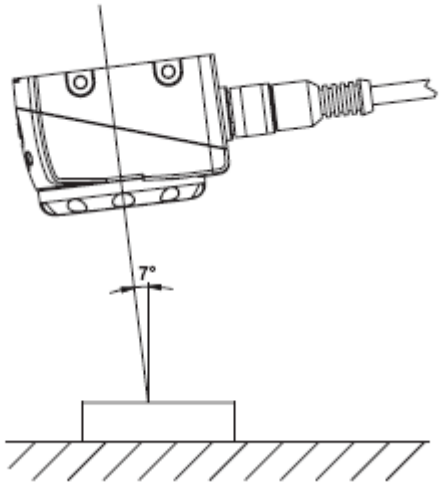
1.2 COMPONENTS

The DataVS2 system includes:

- sensor body with lens and incorporated illuminator;
- configuration software.

2 INSTALLATION

2.1 MOUNTING



The sensor can be mounted with an angle of 5-7° degrees with respect to the vertical axis of the object; in order to avoid strong reflections from the object.

If a 360° Geometric Pattern Match Tool is included in the Inspection performed by the sensor, we recommend to mount the sensor with a 0° angle with respect to the vertical axis of the object. Please refer to par. 5.3.5. for further explanations regarding the 360° Geometric Pattern Match.

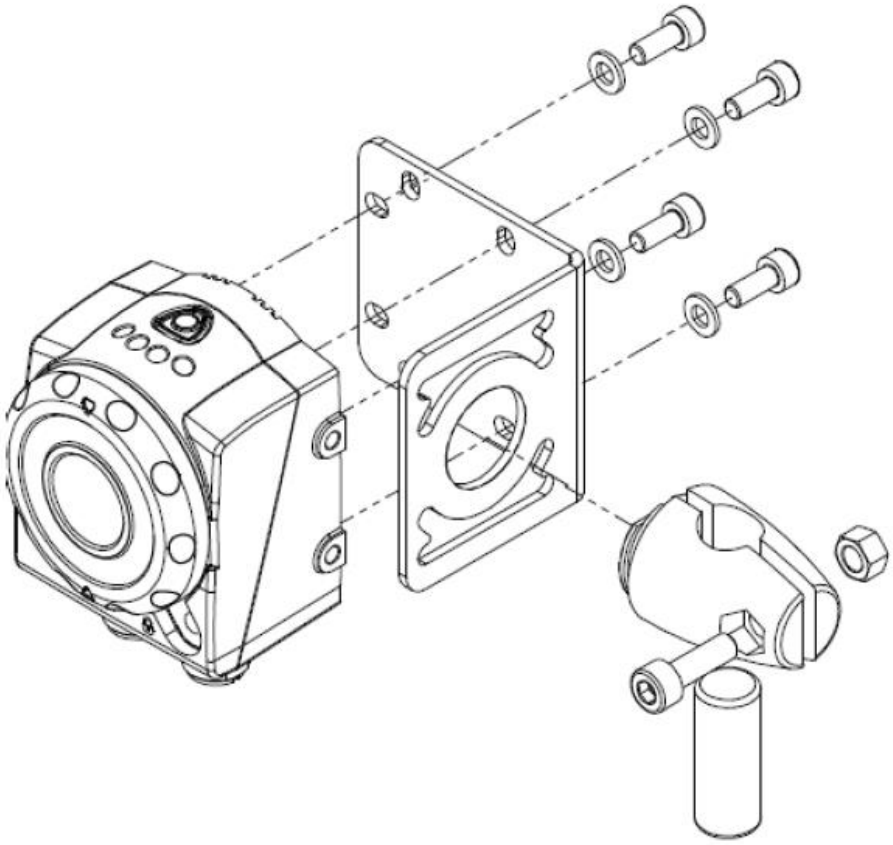
The sensor must be mounted the closest possible to the object, but always respecting the minimum working distance of 50 mm.

Please consider the following points before choosing the working distance: first of all the size of the Field of View (FOV) enhances with the working distance (WD) (see chapter 4.4 for details). Thus you will need to choose a larger WD to get a larger FOV. But the larger the FOV the lower the resolution of the image. Please note that the working distance should be set in order that the camera can detect the characteristics of the object to recognize with as much clarity as possible. Additionally, the longer the working distance the more susceptible the camera becomes to ambient light. We recommend the use of additional lighting for distances greater than 300 mm to reduce the influence of ambient light.

Avoid spilling fluids on the device and make sure it is not exposed to strong vibration.

Do not mount the sensor at a position where the object is exposed to direct sun light or strong ambient light.

To mount the sensor you need four M4 x 6mm screws. The following image shows the mechanical mounting of the sensor using of the mounting kit (see Chapter 15):



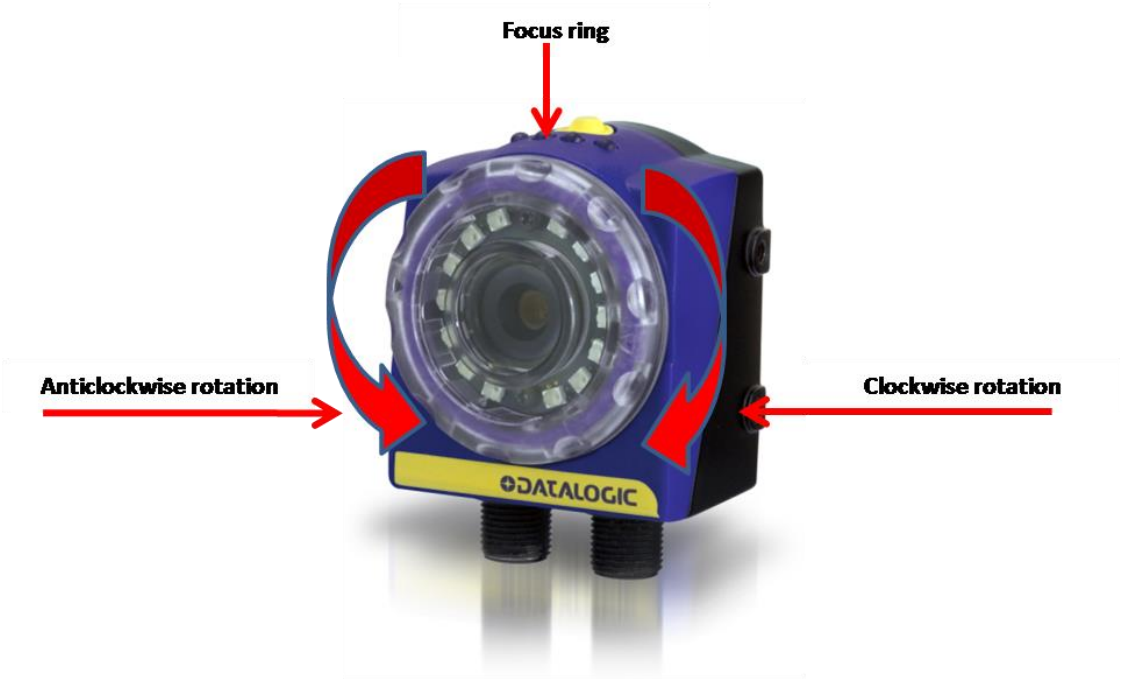
2.2 OPERATING DISTANCES

Working Distance (WD) is the distance from the optical surface of the sensor to the object. The Field Of View (FOV) is the area visible to the sensor at a certain working distance. Normally the size of the FOV increases with the working distance.

Working Distance (mm)	DataVS2-06-DE-XXX	DataVS2-08-DE-XXX	DataVS2-12-DE-XXX	DataVS2-16-DE-XXX	DataVS2-16-DE-XXX-I
50	39 x 29	24 x 18	16 x 12	-	-
80	58 x 44	38 x 29	26 x 19	-	-
110	78 x 58	53 x 40	35 x 26	-	-
140	99 x 73	67 x 50	45 x 34	-	-
170	119 x 88	82 x 61	54 x 41	-	-
200	138 x 104	196 x 72	64 x 48	48 x 36	-
300	207 x 155	144 x 108	96 x 72	72 x 54	72 x 54
400	274 x 206	192 x 144	128 x 96	96 x 72	96 x 72
500	338 x 254	240 x 180	160 x 120	120 x 90	120 x 90
600	405 x 304	288 x 216	192 x 144	144 x 108	144 x 108

2.3 FOCUSING RING

In order to focus the area detected by the sensor you should use the transparent focusing ring on the front side of the sensor.
Referring to the following picture, you must rotate clockwise the focusing ring in order to focus more distant points; you must rotate anticlockwise the focusing ring in order to focus less distant points.



2.4 ELECTRICAL CONNECTIONS

The sensor has two M12 connectors, X1 and X2 (see paragraphs below).
Connect power supply to port X1. To use the sensor without integrating it in the machine environment, you only need to connect PIN2 of Connector X1 to 24 V DC and PIN7 of the connector X1 to Ground.

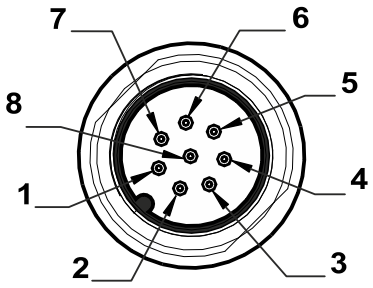
Connector X2 must be connected to an Ethernet 100 Mb port of your PC for sensor configuration through the DataVS2 Graphic User Interface. We recommend to use a “crossed Ethernet cable” for direct connections between the PC and the sensor; a common uncrossed Ethernet cable”, to connect the PC and sensor using the LAN.

To ensure proper operation, never exceed device ratings (see Chapter 12 for a complete list).

NOTE

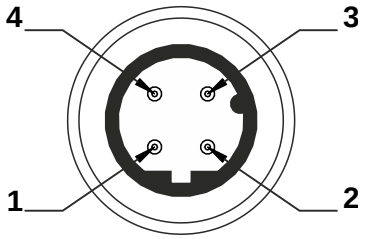
It is not permitted to disconnect the cable at the connector Power Supply and I/O while it is under power.

2.4.1 X1 Connector (M12 8-pin): Power Supply and I/O



PIN	Description		Wire Colour
	DataVS2 OBJ / AOR	DataVS2 ID / PRO	
1	Digital INPUT	RS-232 Rx	White
2	24 V DC	24 V DC	Brown
3	Strobe for external illuminator / OUTPUT 4	Strobe for external illuminator / OUTPUT 4	Green
4	OUTPUT 1	OUTPUT 1	Yellow
5	OUTPUT 2	OUTPUT 2	Gray
6	OUTPUT 3	RS-232 Tx	Pink
7	GND	GND	Blue
8	External trigger (INPUT)	External trigger (INPUT)	Red

2.4.2 X2 Connector (M12 4-pin): Ethernet Connectivity



PIN	Description	Wire Colour
1	Rx+	White/Orange
2	Tx+	White/Green
3	Rx-	Orange
4	Tx-	Green

2.5 GRAPHIC USER INTERFACE INSTALLATION AND START-UP

The supplied DataVS2 Graphic User Interface needs to be installed on your PC to configure the sensor.

2.5.1 Minimum System Requirements

Make sure your personal computer meets the following minimum requirements to ensure proper interfacing with the system:

Component	Recommended	Minimum
Processor(s)	Pentium 4	Pentium 4
Clock frequency	>= 3 GHz	>= 2 GHz
RAM	2 GB	1 GB
Free hard drive space	70 MB	70 MB
Monitor resolution	1280x768	1024x768
Supported Operating System	Windows XP Window Vista Windows 7	

Besides the components listed in the table above, your PC must be equipped with the following hardware and software drivers:

- Installed Ethernet network card and installed driver
- One free 100 Mbps Ethernet port

NOTE

To install and run the DataVS2 Graphic User Interface on Windows Vista / 7 operating systems you need an User Account with Administrator rights.

NOTE

Microsoft Windows XP SP3 is no longer supported: Microsoft no longer provides security updates or technical support for Windows XP.
According to Microsoft, "Running Windows XP SP3 in your environment after April 8, 2014 may expose you to potential risks, such as Security, Compliance, Lack of Independent Software Vendor (ISV) Support and Hardware Manufacturer support.
Datalogic will continue to support existing systems containing Windows XP, but will not be responsible for problems caused by faults in the operating system.
Systems running Windows XP should only connect Ethernet and dataport connections to a network which has routing within the plant or building and no routing outside the plant or building.

.

2.5.2 DataVS2 Graphic User Interface Installation

We recommend closing any programs that are currently running. Then, follow the steps listed below:

1. Download the DataVS2 GUI at www.datalogic.com. Click on **Support** > **Search by product** and enter the DataVS2 family name, then select your product from the dropdown list. Click on the **Software & Utilities** link to download your Graphic User Interface.

2. Run the installation program and follow the installation procedure.

3. Restart the system to complete installation.

4. When the installation is complete the “DataVS2GUI” entry is created in the Start>Programs bar under “Datalogic” as well as a desktop icon. Double-click the desktop icon to run it.

3 SENSOR SETUP

3.1 GENERAL OVERVIEW

The DataVS2 is a sensor acquired images from an object. These images are checked by the means of so called “Vision Controls” and compared with a reference image taken from a reference object. With the “Vision Controls” you can check certain attributes like the contrast between the object surface and the background or the object dimensions.

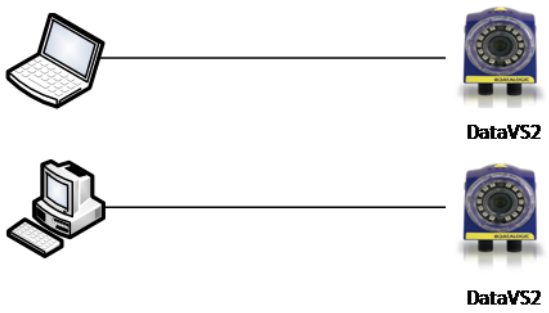
The DataVS2 Graphic User Interface is necessary for sensor adjustment and configuration. It visualizes the images acquired by the sensor and to setup inspections. You can always download the most current version of the DataVS2 Graphic User Interface from the following link: www. datalogic.com.

The figure below gives you an overview of the DataVS2 Graphic User Interface and shows all the sections present.

3.2 COMMUNICATING WITH THE SENSOR

Communication with the sensor occurs via the Ethernet network.
Two types of connection are possible:

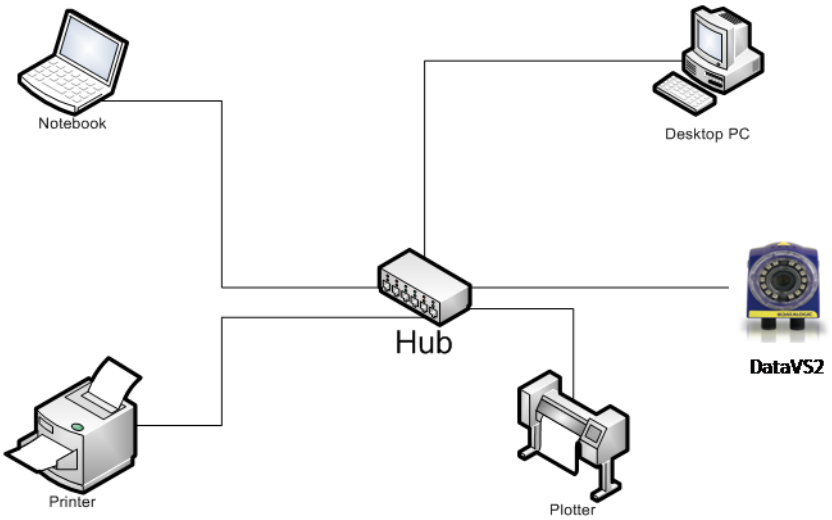
- direct connection: personal computer is connected directly to device using a "cross cable".



- Through LAN: use common network (non-cross) cables normally used to connect devices to routing hubs.

NOTE

This network architecture is denominated "star" configuration because all network cables are connected to a central HUB.



It is essential that the IP addresses of the personal computer and the device are compatible.

The Graphic User Interface offers a discovery feature that automatically searches and locates any sensors in the network and displays their IP address and Subnet Mask information. After selecting the desired sensor, you can set inspection settings (see the following paragraph).

3.2.1 Check PC Network Settings

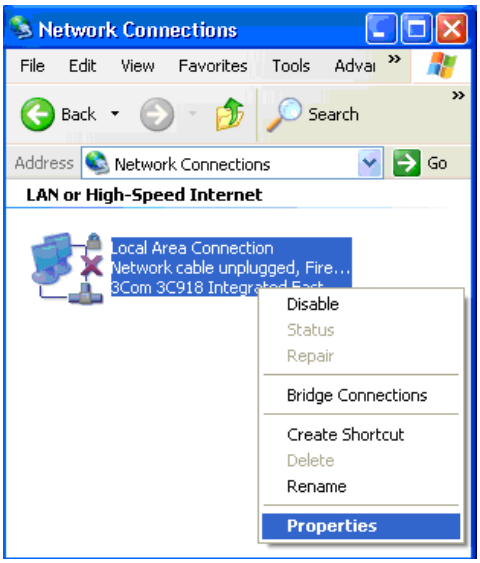
To successfully set up a communication, the IP addresses of the personal computer are compatible with those of the sensor

NOTE

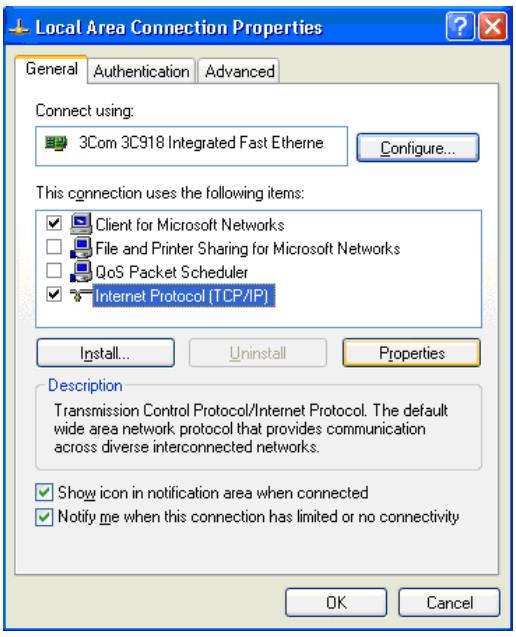
Before running the following instructions, please verify that the user logged in has administrator rights. If this is not the case, please contact your local IT manager for assistance.

If you use a direct connection:

1. From menu START select Control Panel → Network Connections → LAN Connections;
2. Press the right mouse button and select Properties (as shown below);



3. Select Internet Protocol (TCP/IP) then click on the Properties button (as shown below);



4. Select Use the following IP Address option;
5. Enter the following:
   - IP ADDRESS: 172.27.101.1;
   - SUBNET MASK: 255.255.0.0;

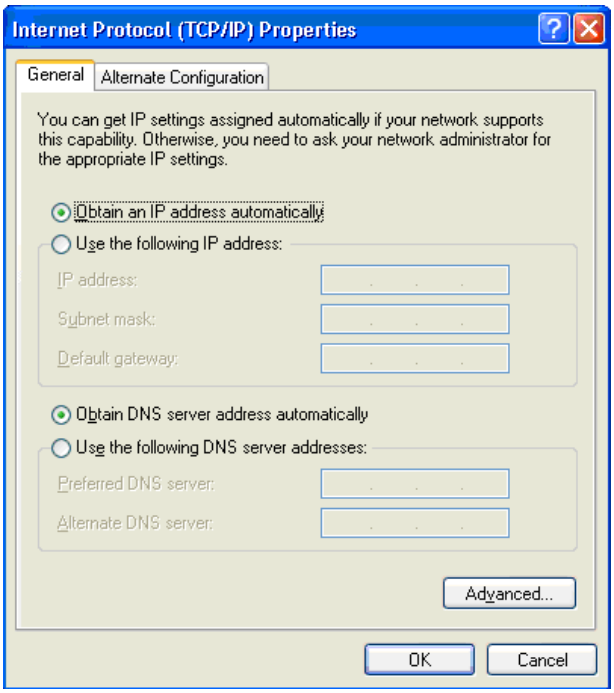
NOTE

No entries for Default Gateway or Use the following DNS Server Address are necessary.

6. Click on OK to close the TCP/IP properties dialog; then click on OK at the LAN Properties dialog.

If you are connected to the sensor through a routing device you should first verify if the network hosts a DHCP server.

- If the network hosts a DHCP Server
  1. From menu START select Control Panel → Network Connections → LAN Connections;
  2. Press the right mouse button and select Properties;
  3. Select Internet Protocol (TCP/IP) then click on Properties button again;
  4. Finally select Obtain an IP Address automatically (as shown below).



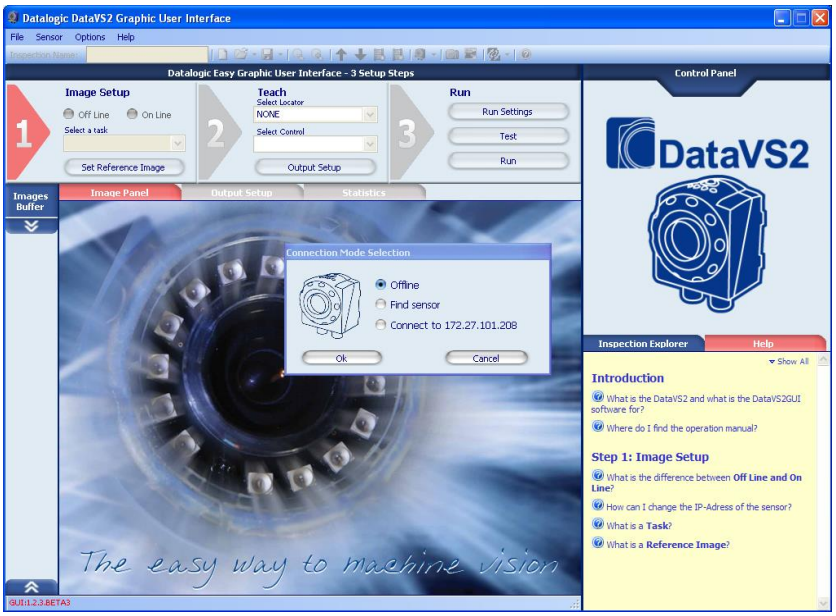
Depending on the settings of your PC the current status of the network connection is shown in the task bar. Please wait until the network connection is established before connecting your sensor.

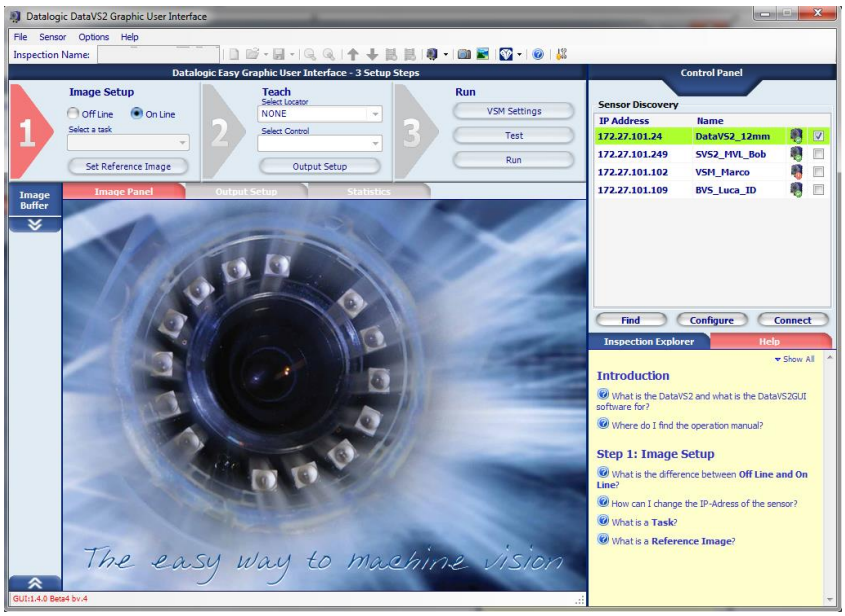
- If the network doesn't host a DHCP Server:
  1. Please contact your local IT manager for assigning an IP Address to your PC;
  2. Follow the instructions for direct connection configuration.

3.2.2 Set up Communication

To setup a communication between sensor and the DataVS2 GUI please follow this list of instructions:

1. Connect PIN2 of Connector X1 to 24 V DC; connect PIN 7 of Connector X1 to Ground (see Chapter 2 for electrical connection description).
2. Disconnect any Ethernet cable connected to your PC
3. Connect Connector X2 to Ethernet 100 port of the PC
4. Start the DataVS2 Graphic User Interface
5. Select in STEP1: “Online”, then “Find sensor” (see picture below). The software now shows all found sensors at the right side of the screen, in the CONTROL PANEL. Select which sensor on the network to connect to in order to establish communication.





6. Click on the “Connect” button and the software will try to establish a connection with the sensor: if success, a message will appear, displaying the currently connected sensor Model Type and the related Firmware version.

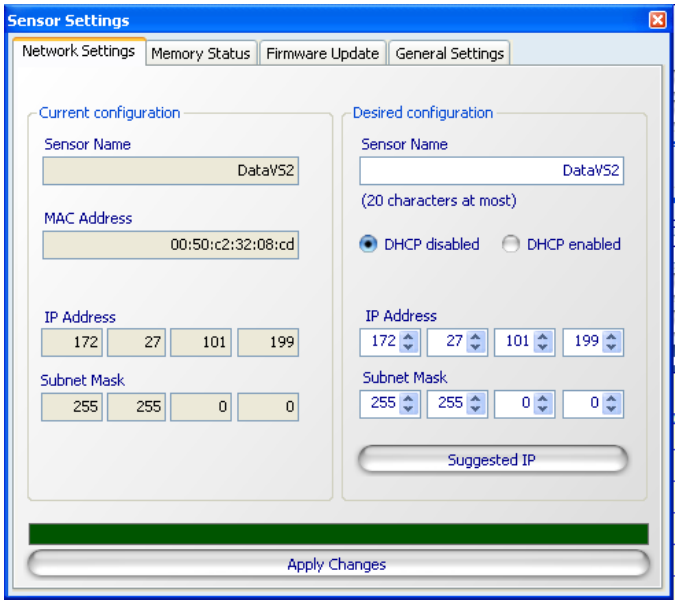
You have successfully established communication and may proceed to setting your inspection.

NOTE *"Find sensor" is a discovery feature that automatically searches and locates any sensors in the network and displays their IP addresses and Subnet Mask information. It is based on Broadcast protocol and can work only inside a Local Area Network*

3.2.3 Set up Sensor IP-Address

As a default all sensors have this IP-Address: 172.27.101.208. The DataVS2 Graphic User Interface allows to set up an individual IP-Address for every sensor. Please follow these steps if you want to change the IP-Address:

1. Connect to the sensor as described in the paragraph above.
2. Select the menu "Sensor" from the menu bar, select the entry "Settings" and finally select the entry "Edit". The DataVS2 GUI will open a dialog displaying different Tabs for Sensor Parameters configuration. Select the Network Settings Tab:



On the left side of the Network Settings Tab you will see the current sensor configuration:

- The current name of the sensor which can be 20 characters long
- The current IP-Address of the sensor
- The current Subnet Mask

The right side of the Network Settings Tab will allow you to change the Name, the IP-Address and the Subnet Mask of the sensor.

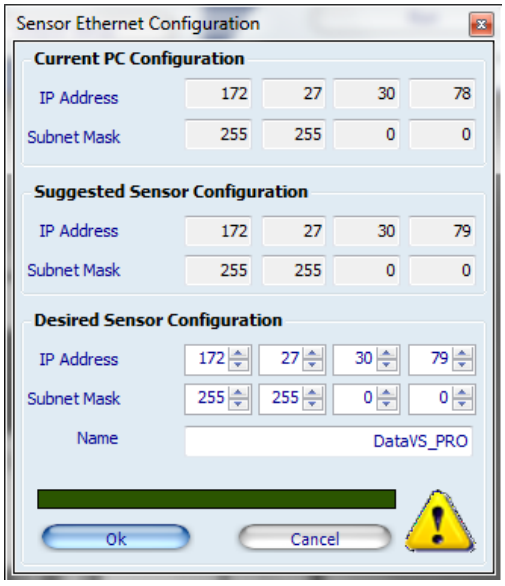
Please consider the following notes:

- The name can have only 20 characters. All characters shall be plain ASCII.
- Option “DHCP enabled” will launch the DHCP client process running on your sensor. After power off and restart the sensor IP-Address will automatically be configured by the DHCP Server process running on your LAN.
- If your LAN doesn’t host a DHCP server process or if you want to assign the sensor a static IP-Address, please choose the option “DHCP disabled” and insert a valid IP-Address and Subnet Mask. You may also press button “Suggested IP”: the DataVS2 GUI will suggest you a valid IP-Address for the sensor.
- The new IP-Address shall differ from those your PC is currently using. Otherwise the sensor cannot be connected any more with the PC.
- The new IP-Address shall not be set to “0.0.0.0”. Otherwise communication with the sensor will be impossible.
- The new Subnet Mask shall not be set to “0.0.0.0”. Otherwise communication with the sensor will be impossible.
- The new IP-Address shall not be set to 127.0.0.1. This IP-Address is normally used for loopback communications between processes running on the same PC.

3. After you have clicked “Apply changes” button the sensor will store the new configuration. The DataVS2 GUI will disconnect from sensor. The new IP-Address will be set active after the next POWER UP of the sensor. The old settings remain active until next POWER UP.
   If the fw sensor is ST2.4.0.XX or higher, the sensor will reboot automatically.

An alternative way to change the sensor IP-Address is the following one. You can use it when no communication is established between sensor and PC:

1. Select “Online” button in STEP1. A “connection mode” pop-up window will open.
2. Select “Find sensor”. DataVS2 GUI searches for all connected sensors and shows all found sensors in the so called CONTROL PANEL.
3. Select from the list the sensor whose IP-Address is to change.
4. Select “Configuration” button. The following window will appear:



In the top section you find the current IP-Address of the PC. In the middle you find a suggestion on the network settings of the sensor. In the lower section you find the input boxes for the network settings (use arrow keys, numerical entries and the scroll of the mouse-wheel). After clicking on “Ok” the IP-Address will be sent to and stored by the sensor. The new IP-Address will not become active until the next power-up of the sensor.
If the fw sensor is ST2.5.0.XX or higher, the sensor will reboot automatically.

3.2.4 Troubleshooting

If the connection between the sensor and the software does not work immediately, please check the following issues:

- Check the wiring of the sensor to the power supply and the communication cable.
- Check the connection as described in Chapter 3.2.1.
- Exit the Software and restart it again.
- If you have a Firewall running on your PC: please check if the TCP port 5423 is enabled and not used by any other program. If not enabled, please enable this port.
- Run the Recovery mode (see Chapter 10 in this manual) and follow the described steps to verify the sensor works with the initial parameters.
- After you have done this, exit the Software and restart both the sensor and the Software again. If the sensor still not work, try to connect another sensor with the PC to verify if the sensor interface has been damaged.

3.3 UPDATING SOFTWARE

Every sensor comes with the most current DataVS2 GUI software available.
Datalogic will supply updates to this software on its website to provide you with enhanced features and new functions.

If you want to update the DataVS2 GUI software simply follow the instructions below:

1. Uninstall the software (if installed on your PC) using Microsoft Windows Software dialog
2. Press the Windows START button
3. Select CONTROL PANEL → System → Software
4. Select DataVS2 GUI out of the list of installed software and press the “Delete” button shown at the right side of the screen: Windows now uninstalls the DataVS2 GUI software.

Images and projects are not deleted with this procedure.

NOTE

5. Download the new software version from Datalogic website, if not already done
6. Extract the installation files from the zip package
7. Select the folder where you saved the downloaded files and extract them; double click on the “DataVS2GUI.exe”: the update of the DataVS2 GUI software is now installed.

3.4 GENERAL SENSOR SETTINGS

The following table displays all the sensor parameters that can be configured through the DataVS2 GUI software.

The Firmware Version column represents for each parameter all the sensor firmware versions that are able to manage the corresponding functionality; as an example, the DHCP functionality is not managed by 0.0.X firmware version. This means that when the DataVS2 GUI software is connected to a sensor running a 0.0.X firmware version the DHCP controls will be disabled.

The Factory Settings represent for each parameter the default value that you will find on your sensor at the first connection.

The values column contains all the admitted values for the corresponding parameter.

The DataVS2 OBJ, AOR, ID and PRO columns refer to the sensor Model Type and are checked depending on the availability of the parameter in the selected Model.
When these columns are not present it means that the parameter is available for each model type.

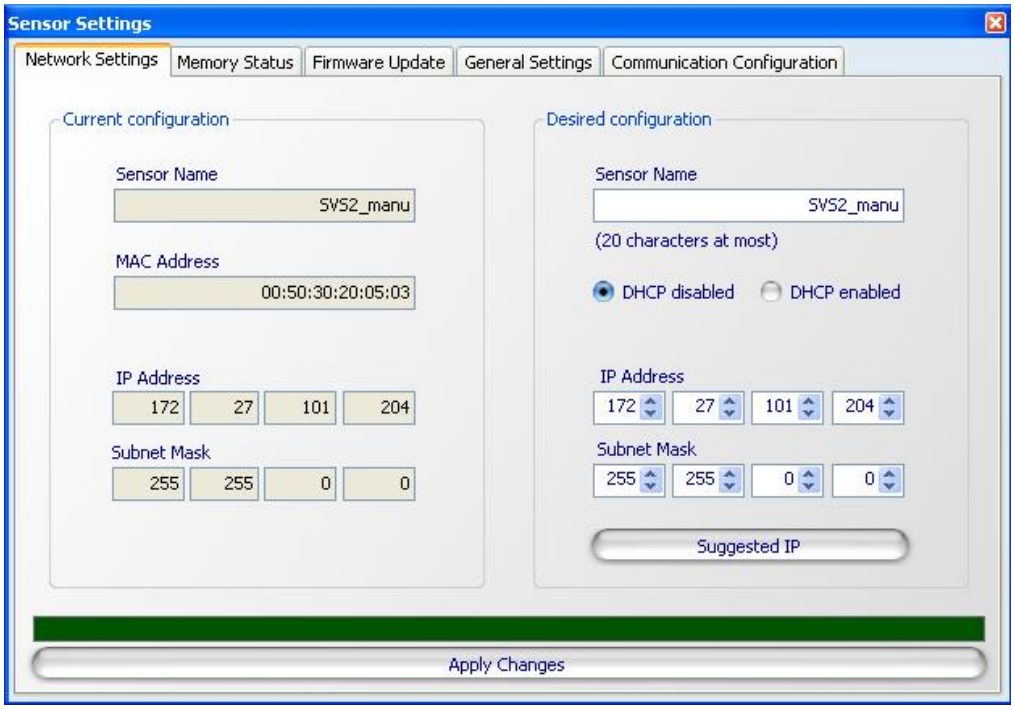
Parameters	Firmware version	Default	Values	OBJ	AOR	ID	PRO
Sensor name	0.0.X	DataVS	• 4 ASCII characters for 0.0.X versions • 20 UTF-8 characters for ST2.X.Y versions	☒	☒	☒	☒
Currently running inspection	0.0.X	1	• [1-8] for 0.0.X versions • [1-20]: for ST2.X.Y versions	☒	☒	☒	☒
Number of inspections stored on the sensor	0.0.X	1	• [1-8] for 0.0.X versions • [1-20]: for ST2.X.Y versions	☒	☒	☒	☒
IP address	0.0.X	172.27.101.208		☒	☒	☒	☒
Subnet mask	0.0.X	255.255.0.0		☒	☒	☒	☒
Insp Switching pulse duration	0.0.X	10ms	[10-100] ms	☒	☒	-	-
Current Insp Switching protocol	ST2.1.X	Standard	• Standard • Expert • Teach in • Expert + Teach in	☒	☒	-	-
DHCP	ST2.1.X	Disabled	• Enabled • Disabled	☒	☒	☒	☒
Clock Mode	ST2.1.X	180MHz OBJ 480MHz AOR, ID, PRO	• 180MHz • 480MHz	-	☒	☒	☒

Parameters	Firmware version	Default	Values	OBJ	AOR	ID	PRO
Trigger Ethernet / Serial String	ST2.3.0.X	“TRIGGER”	• 7 bytes with ASCII coding or belonging to the [0, 31] value range + terminator (corresponding to NULL).	☒	☒	☒	☒
Serial Results Prologue String	ST2.3.0.X	<STX>	• 7 bytes with ASCII coding or belonging to the [0, 31] value range + terminator (corresponding to NULL).	-	-	☒	☒
Serial Results Epilogue String	ST2.3.0.X	<CR><LF>	• 7 bytes with ASCII coding or belonging to the [0, 31] value range + terminator (corresponding to NULL).	-	-	☒	☒
Ok	ST2.4.0.Y	Enabled	• Enabled • Disabled	☒	☒	☒	☒
Length	ST2.4.0.Y	Enabled	• Enabled • Disabled	☒	☒	☒	☒
Serial Communication Protocol	ST2.4.0.X	Synchronous	• Synchronous • Asynchronous	-	-	☒	☒
RS-232: Baud Rate	ST2.3.0.X	57600 bit/s	• 1200 • 2400 • 4800 • 9600 • 19200 • 57600 • 115200	-	-	☒	☒
RS-232: Number of Bits per Datum	ST2.3.0.X	8	• 5 • 6 • 7 • 8	-	-	☒	☒
RS-232: Parity Type	ST2.3.0.X	NONE	• NONE • EVEN • ODD	-	-	☒	☒
RS-232: Number of Stop Bits	ST2.3.0.X	1	• 1 • 2	-	-	☒	☒

To configure sensor settings the DataVS2 GUI software presents a special window, denominated “Sensor Settings”, that can be opened from the Sensor menu. The functions available in “Sensor Settings” window are organized on 5 tabs, explained in the following paragraphs.

3.4.1 Network Settings

This panel allows you to setup the sensor Ethernet communication parameters. The window is organized into two areas: “Current Configuration”, representing the current Ethernet configuration on the sensor and “Desired Configuration”.

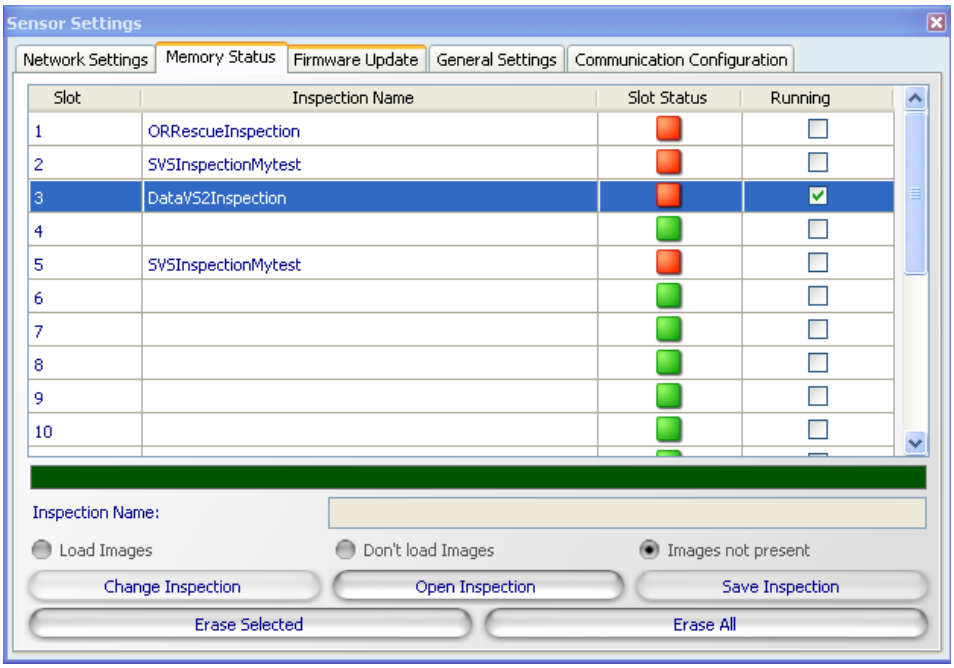


Parameters	Description
Sensor name	The current sensor and desired name. To change the name of the current sensor enter the desired one and press the “Apply Changes” button. ***Note***: the sensor name must be a non-empty string composed of maximum 20 characters. ***Note***: sensor reset is required after changing this value if the sensor fw version is ST2.3.0.XX or older.
DHCP disabled	Disables the DHCP function. ***Note***: this control is disabled for sensors running on 0.0.X firmware version. ***Note***: sensor reset is required after changing this value.
DHCP enabled	Enables the DHCP function. With the DHCP function enabled, the sensor is configured with a dynamic IP address retrieved by a DHCP server installed on your LAN. While waiting for the IP address from the DHCP server the sensor set the LED1, LED2 and LED4 on. If no DHCP server is running on your network the sensor will restore the static IP Address 1 minute after timeout. With DHCP enabled you don’t have to configure the sensor’s IP Address any more. ***Note***: this control is disabled for sensors running on 0.0.X firmware version. ***Note***: DHCP function will enable the DHCOP Client process on the sensor. Be sure your network is running on a DHCP server before enabling this function on the sensor, otherwise the sensor will not be able to get a dynamic IP Address. ***Note***: sensor reset is required after changing this value if the sensor fw version is ST2.3.0.XX or older.
IP address	The current sensor IP Address. Use this field to manually change the sensor’s IP Address. ***Note***: this function is disabled when the DHCP option is enabled. ***Note***: sensor reset is required after changing this value if the sensor fw version is ST2.3.0.XX or older.

Parameters	Description
Subnet Mask	The current sensor Subnet Mask. Use this field to manually change the sensor's Subnet Mask. ***Note***: this function is disabled when the DHCP option is enabled. ***Note***: sensor reset is required after changing this value if the sensor fw version is ST2.3.0.XX or older.
Suggest IP	After pressing this button the DataVS2 GUI will calculate a valid, static IP Address for the sensor and display it in the two fields above the button. We recommend using this function when you need to change the IP Address of the sensor. ***Note***: this function is disabled when the DHCP option is enabled.
Apply changes	Press this button to apply the desired Configuration on the sensor. ***Note***: After applying the changes on the sensor, DataVS2 GUI will disconnect itself from the sensor and you must reset the sensor.

3.4.2 Memory Status

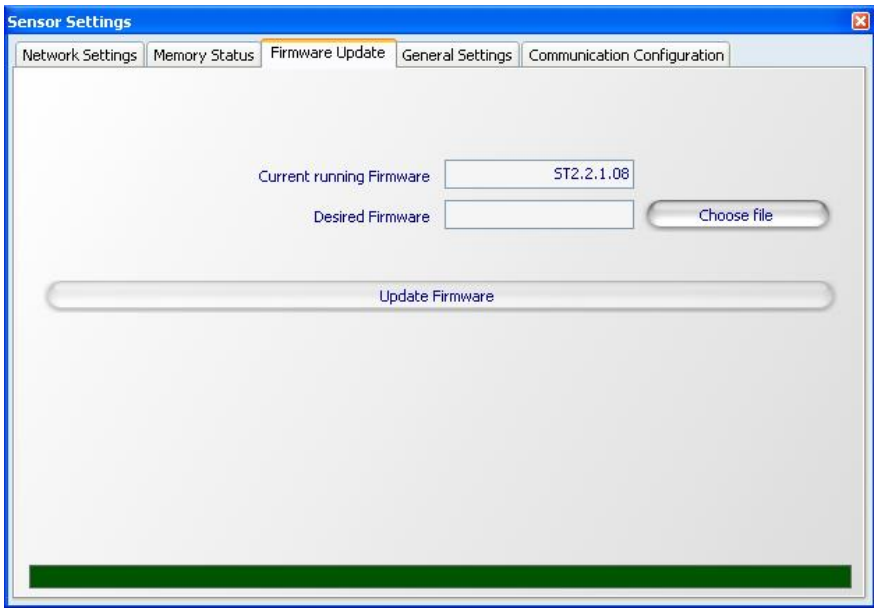
This panel allows you to configure the sensor's memory. It is also used for "Opening Inspection from sensor" and "Saving Inspection on sensor" Tasks.



Parameters	Description
Slot Column	The sensor's memory Slot. The sensor's memory dedicated to Inspection storage is organized in Slots. A Slot is a sensor memory area where you can store an Inspection. The sensor's ST2.1.X firmware version has 20 memory slots available. Older firmware versions have only 8 memory slots.
Inspection Name Column	The name of the Inspection stored in the corresponding memory slot.
Slot Status Column	Indicates if a Slot is empty (green) or not (red).
Current Column	Indicates the current Inspection running on the sensor. ***Note***: only one Inspection at a time can be checked as running. ***Note***: if an empty Slot is checked as Running, the sensor is in an IDLE status, i.e. it doesn't execute any Inspection.
Text box Inspection name	The name of the Inspection to be saved on the sensor. ***Note***: this field is enabled only when performing a "Save Inspection on sensor" task.
Load Images	Enabled only when performing a "Load Inspection from sensor" task. If images are present on the sensor (stored during Inspection execution) select this option to load images together with the Inspection on DataVS2 GUI.
Don't load images	Enabled only when performing a "Load Inspection from sensor" task. If images are present on the sensor (stored during Inspection execution) select this option to avoid loading images together with the inspection on DataVS2 GUI.
Images not present	This option is selected when performing a "Load Inspection from sensor" task and no images are present on the sensor.
Change Inspection	Press this button to change the currently running Inspection on the sensor. After pressing this button, the Inspection corresponding to the selected table row will be selected as the currently running one. ***Note***: the selected Slot is highlighted in blue.
Open Inspection	Press this button to "Load an Inspection from Sensor". After pressing this button the Inspection corresponding to the selected row will be loaded and shown by the DataVS2 GUI. ***Note***: the selected Slot is highlighted in blue.
Save Inspection	Press this button to "Save Inspection on sensor". After pressing this button the current Inspection will be saved in the selected memory slot. ***Note***: the selected Slot is highlighted in blue.
Erase Selected Slot	Clear the selected memory Slot. The Inspection will be deleted. ***Note***: the selected Slot is highlighted in blue.
Erase all	Clear all the memory Slots. All the Inspections currently stored on the sensor will be deleted.

3.4.3 Firmware Update

This panel allows you to perform a Firmware Update operation.



Parameters	Description
Currently running firmware	The version of the firmware currently running on the connected sensor.
Choose File	Press this button to select the desired firmware file to upload on the sensor. A dialog box will be opened allowing you to choose the desired firmware file stored on your PC. Available firmware files can have two different extensions: **sfw2:** denotes a STD.2.1.X firmware file, updated to the last revision. **sfw**: denotes a 0.0.X firmware file, an older version. After selecting the desired firmware file and pressing the “OK” button from the dialog box, the selected firmware version will be displayed in the “Desired Firmware” field.
Update Firmware	Press this button to update firmware on the connected sensor. Two types of firmware updates are available: **firmware upgrade:** a newer firmware version is uploaded on the sensor. **firmware downgrade**: an older firmware version is uploaded on the sensor. ***Note*:** beginning from DataVS2 GUI 1.3.0 release, firmware update will memorize all the Inspections stored on the sensor in the DataVS2 GUI project directory and delete them from the sensor. At the end of the updating, all the sensor Inspections have to be reset manually. ***Note***: if this button is disabled, the selected firmware is not compatible with the connected sensor due to hardware version incompatibility. In this case, please select a different firmware. ***Note***: After updating sensor firmware, DataVS2 GUI will disconnect itself from the sensor and you must reset the sensor.

Firmware Update Table

The following table describes the sensor's behaviour in the different Firmware update situations during the following operations:

- [Backup] Backup of Inspections stored in the sensor
- [Del] Deletion of Inspections stored in the sensor
- [Updt Std Fw] Update of Standard Firmware currently stored in the sensor

Current Firmware	**New Firmware**				
	0.0.Y	ST2.2.1.XY	ST2.3.0.XY	ST2.4.0.XY	ST2.5.0.XY
0.0.X	Backup: YES Del: NO Updt Std Fw: YES	Backup: YES Del: YES Updt Std Fw: YES	Backup: NO Del: NO Updt Std Fw: NO	Backup: NO Del: NO Updt Std Fw: NO	Backup: NO Del: NO Updt Std Fw: NO
0.0.X RESCUE	Backup: YES Del: YES Updt Std Fw: YES	UPDATE NOT PERMITTED	UPDATE NOT PERMITTED	UPDATE NOT PERMITTED	UPDATE NOT PERMITTED
ST2.2.1.XX	Backup: YES Del: YES Updt Std Fw: YES	Backup: YES Del: NO Updt Std Fw: YES	Backup: YES Del: YES Updt Std Fw: YES	Backup: YES Del: YES Updt Std Fw: YES	Backup: YES Del: YES Updt Std Fw: YES
ST2.2.1.XX RESCUE	UPDATE NOT PERMITTED	Backup: YES Del: YES Updt Std Fw: YES	UPDATE NOT PERMITTED	UPDATE NOT PERMITTED	UPDATE NOT PERMITTED
ST2.3.0.XX	Backup: NO Del: NO Updt Std Fw: YES	Backup: YES Del: YES Updt Std Fw: YES	Backup: YES Del: NO Updt Std Fw: YES	Backup: YES Del: YES Updt Std Fw: YES	Backup: YES Del: YES Updt Std Fw: YES
ST2.3.0.XX RESCUE	UPDATE NOT PERMITTED	UPDATE NOT PERMITTED	Backup: YES Del: NO Updt Std Fw: YES	UPDATE NOT PERMITTED	UPDATE NOT PERMITTED
ST2.4.0.XX	Backup: NO Del: NO Updt Std Fw: YES	Backup: YES Del: YES Updt Std Fw: YES	Backup: YES Del: YES Updt Std Fw: YES	Backup: YES Del: NO Updt Std Fw: YES	Backup: YES Del: YES Updt Std Fw: YES
ST2.4.0.XX RESCUE	UPDATE NOT PERMITTED	UPDATE NOT PERMITTED	UPDATE NOT PERMITTED	Backup: YES Del: NO Updt Std Fw: YES	UPDATE NOT PERMITTED
ST2.5.0.XX	Backup: NO Del: NO Updt Std Fw: YES	Backup: YES Del: YES Updt Std Fw: YES	Backup: YES Del: YES Updt Std Fw: YES	Backup: YES Del: YES Updt Std Fw: YES	Backup: YES Del: NO Updt Std Fw: YES
ST2.5.0.XX RESCUE	UPDATE NOT PERMITTED	UPDATE NOT PERMITTED	UPDATE NOT PERMITTED	UPDATE NOT PERMITTED	Backup: YES Del: NO Updt Std Fw: YES

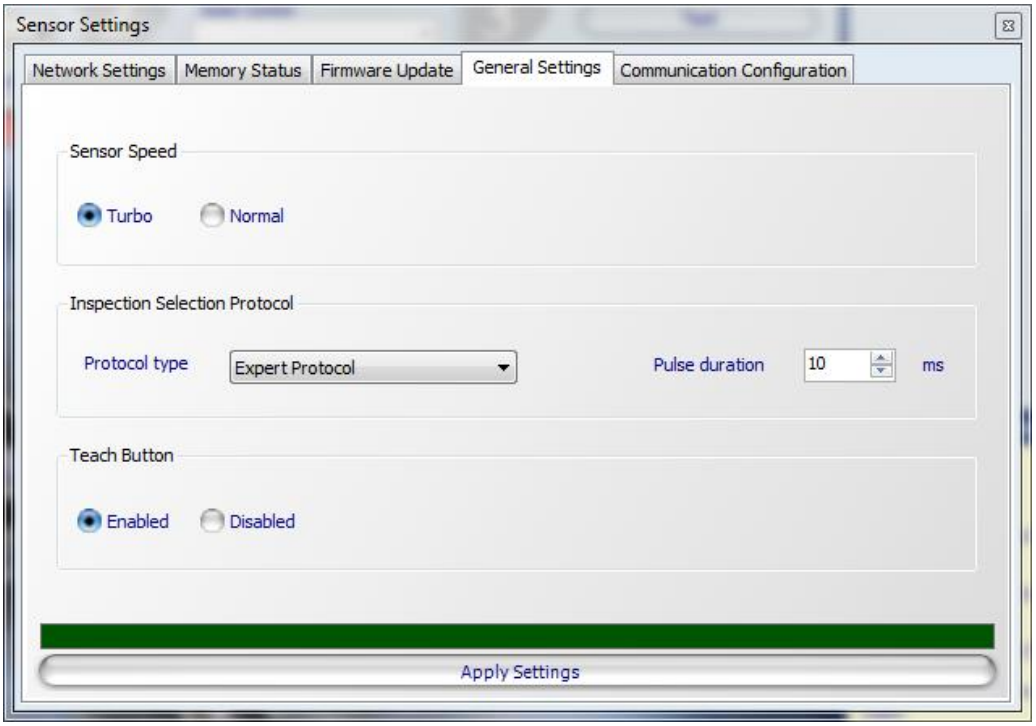
3.4.4 Firmware Update Rules

The sensor's behaviour during firmware updating is controlled by a few rules:

- In the RESCUE mode, the user can load only one firmware belonging to the same class of versions. The RESCUE firmware is not changed
- FW update (both UPGRADE and DOWNGRADE) always generates:
  - backup of the Inspections

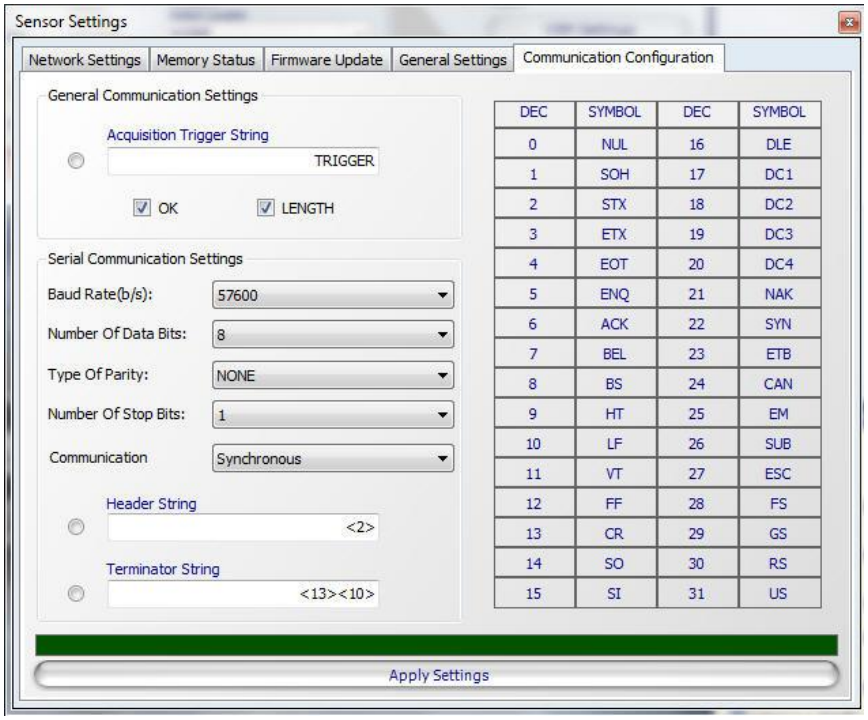
  - deletion of all the memory slots
  - FW update
- The user cannot directly go from 1.1.4 to 1.3.0 or vice versa
- The following operations have to be carried-out to upgrade from ST2.2.1.XY firmware version to ST2.3.0.XY:
  1. Update the sensor to ST2.3.0.XY.BRIDGE firmware version
  2. Restart the sensor
  3. Update the sensor to ST2.3.0.XY firmware version

3.4.5 General Settings



Parameters	Description	OBJ	AOR	ID	PRO
Turbo Sensor Speed	Set the connected sensor clock frequency to 480 MHz.	-	☒	☒	☒
Normal Sensor Speed	Set the connected sensor clock frequency to 180 MHz.	☒	☒	☒	☒
Standard Protocol	Enables the digital Inspection selection and selects the “Standard Protocol” as the current digital Inspection selection protocol.	☒	☒	-	-
Expert Protocol	Enables the digital Inspection selection and selects the “Expert Protocol” as the current digital Inspection selection protocol.	☒	☒	-	-
Expert Protocol plus Teach In	Enables the digital Inspection selection, selects the “Expert Protocol” as the current digital Inspection selection protocol and enables the digital Teach In function.	☒	☒	-	-
Teach In	Enables the digital Teach In function. Disables the digital Inspection selection function.	☒	☒	-	-
Digital Pulse Duration	Represents the minimum duration required for digital I/O pulses used in the Inspection switching protocol (see Chapter 6).	☒	☒	-	-

3.4.6 Serial Communication and Ethernet Parameters



Parameters	Description	OBJ	AOR	ID	PRO
Trigger String	Represents the character string accepted as Trigger by the sensor when an Inspection is running with Trigger Mode set on the “Ethernet / Serial” value. To set this value, the radio button corresponding to “Trigger String” text field has to be selected and the desired string has to be entered if different from the one currently displayed. The string must have maximum 7 bytes corresponding to ASCII characters or non-printable characters (values belonging to the [0, 31] range). To enter one or more non-printable characters the user has to simply select, with the mouse, the decimal value corresponding to the desired symbol in the table visualized on the right side of the window. ***Note***: the sensor accepts the Trigger command on serial communication only if it is composed of a string of characters (set by the interface) that ends with the <NULL> symbol (represented by byte 0).	☒	☒	☒	☒
Ok	Represents if the “OK” string is present in the response to the GETSTRINGRESULTS request.	☒	☒	☒	☒
Length	Represents if the string with the length of the message is present in the response to the GETSTRINGRESULTS request.	☒	☒	☒	☒
Results Prologue String	Represents the character string that composes the prologue of each data package sent by the sensor to the Serial port. To set this value, the radio button corresponding to “Prologue” text field has to be selected and the desired string has to be entered if different from the one currently displayed. The string must have maximum 7 bytes corresponding to ASCII characters or non-printable characters (values belonging to the [0,	-	-	☒	☒

Parameters	Description	OBJ	AOR	ID	PRO
	31] range). To enter one or more non-printable characters the user has to simply select, with the mouse, the decimal value corresponding to the desired symbol in the table visualized on the right side of the window.				
Results Epilogue String	Represents the character string that composes the epilogue of each data package sent by the sensor to the Serial port. To set this value, the radio button corresponding to “Epilogue” text field has to be selected and the desired string has to be entered if different from the one currently displayed. The string must have maximum 7 bytes corresponding to ASCII characters or non-printable characters (values belonging to the [0, 31] range). To enter one or more non-printable characters the user has to simply select, with the mouse, the decimal value corresponding to the desired symbol in the table visualized on the right side of the window.	-	-	☒	☒
RS-232: Baud Rate	Represents the transmission speed, expressed in bits per second, on the RS-232 serial channel.	-	-	☒	☒
RS-232: Number of Bits per Datum	Represents the number of bits that composes a word on the RS-232 serial channel.	-	-	☒	☒
RS-232: Parity Type	Represents the optional bits used to check the integrity of the word transmitted on the RS-232 serial channel.	-	-	☒	☒
RS-232: Number of Stop Bits	Represents the number of bits used to separate two consecutive words on the RS-232 serial channel.	-	-	☒	☒
Serial Communication Protocol	Defines the communication protocol used for sending the inspection results on the Serial communication channel: **Synchronous**: DataVS2 sends a result message after receiving a request from a connected client (GETSTRINGRESULTS command, Chapter 8); ***Note***: all Inspection results may not be communicated to the client in this mode. This depends on the faster inspection speed of DataVS2 with respect to the sending frequency of the client’s requests. **Asynchronous**: DataVS2 sends the inspection results at the end of each inspection cycle without waiting for the connected client’s request. ***Note***: this function is available in ST2.4.0.01 firmware, and latter versions, of DataVS2.	-	-	☒	☒

NOTE

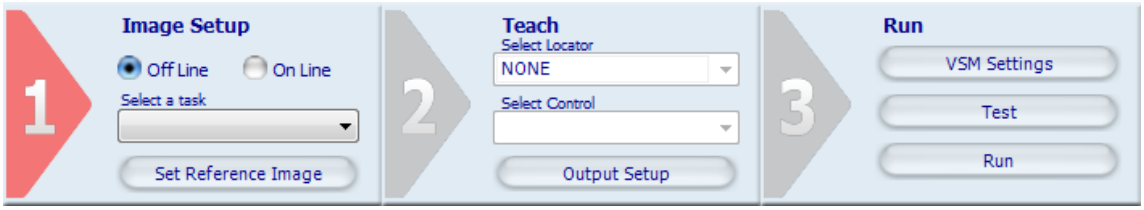
The parameters present in the “Communication configuration” panel are enabled only when DataVS2 GUI is connected to a DataVS2 AOR, ID or PRO sensor.

NOTE

The “Serial setup” parameters present in the “Communication configuration” panel are enabled only when DataVS2 GUI is connected to a DataVS2 ID or PRO sensor.

4 INSPECTION SETUP

To operate as a stand-alone device the DataVS2 Vision Sensor requires a preliminary configuration procedure. The result is a so called Inspection. The term Inspection represents a certain set of operations the sensor performs in order to check the features of an object. If all features comply with certain specifications configured during Inspection setup, the Inspection result will be SUCCESS, otherwise the Inspection result will be FAILURE. The DataVS2 GUI software will lead you through three STEPS needed for building an Inspection. Each STEP is identified by a number. When a STEP is active the DataVS2 GUI will display it in a bright red highlight; inactive STEPS are shown with blue colour. If you need to change the parameters already configured during setup procedure, you have to simply click on the triangle of the STEP you want to change to. After clicking it, this STEP will be highlighted.



- **STEP1: Image Setup.** Allows you to choose operation mode, create or load an Inspection and setup image acquisition properties.
- **STEP2: Teach.** Allows you to insert Locators, Controls and configure Output behaviour.
- **STEP3: Run.** Allows you to set the Inspection parameters that can be changed by the VSM (Vision Sensor Monitor), test the Inspection and finally run it on the sensor.

4.1 GENERAL CONSIDERATIONS

A successful Inspection is based on a reference image with good quality, as the image of each object to be inspected will be compared with it. The object features you want to inspect should be clearly visible in the reference image. For this reason you should assure the reference image is characterized by the following properties:

- assure the reference image is in focus, with sharp clear contours
- assure the reference image is characterized by high contrast between the target object and the background.

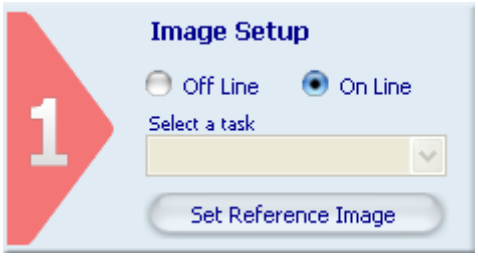
A few tips are listed below to facilitate a correct configuration of the sensor:

- determine which product feature you want to monitor
- provide correct, even lighting
- minimize the influence of the ambient light on the target
- accurately choose the Controls to be used for the Inspection in order to provide the best result
- Accurately setup Control parameters and tolerances: some features may be hard to identify with low tolerances.

The majority of parts considered "good" never has a 100% match with the reference image acquired. Position offset or light level fluctuations can affect the match score. Any "bad" part should differ from "good" parts in as many characteristics as possible to obtain large gap between the match score of the good and the bad parts. We recommend trying several good parts while testing your application as well as several bad parts. Be sure to consider eventual changes in lighting conditions and position as well when you choose the appropriate Desired

value. The following paragraphs will describe the single STEPS, relative options and selectable parameters.

4.2 STEP 1: IMAGE SETUP



During this first step, the DataVS2 GUI allows you to perform the following operations:

- Select the operation mode of DataVS2 GUI, either Online or Offline (Simulator mode).
- Select the current Task: create a new inspection or change an existing one.
- Setup the basic settings of the Vision Sensor while in Online mode.
- Select the reference image you want to build your inspection with.

4.2.1 Quick Guide STEP 1

NOTE

Please, follow the steps described in Chapter 3, Setup Communication, before proceeding with the following ones.

1. Select “Online” button in STEP1; the DataVS2 GUI software visualize the “Connection Mode Selection” window.
2. Select “Find sensor”. The software now shows all sensors found at the right side of the screen, in the CONTROL PANEL. Select which sensor on the network to connect to in order to establish communication.
3. Click on the “Connect” button and the software will try to establish a connection with the sensor: if successful, a message will appear, displaying the currently connected sensor Model Type and the related Firmware version.

NOTE

After the connection to the sensor is established, if no Task is currently selected (STEP 1 and STEP 2 triangles are disabled and displayed in gray), the DataVS2 GUI automatically selects a “New Inspection” Task and starts displaying images acquired by the sensor in the IMAGE PANEL. If a Task is already selected in STEP 1, from the drop-down menu below the “Select a Task” label choose “New Inspection”; the DataVS2 GUI will now visualize the “Model Type Selection” window: choose the same sensor Model as your sensor one and press “Ok” button.

The DataVS2 GUI will now display “Basic Settings” parameters at the right side of the main window, in the CONTROL PANEL.

4. Set Internal Illuminator to “On”.
5. Press “Auto exposure” button and change Gain value till the image has a good contrast and brightness level.

6. Manually adjust the focus of the sensor by means of the sensor's front side focus ring until a sharp picture is seen. If the current image is sharp but has poor contrast you should repeat the previous steps.
7. Click on "Set Reference Image" button; now you have selected the currently displayed image as reference for your Inspection.

4.2.2 Operating Modes

The DataVS2 GUI offers two different operating modes:

- **Online mode**, with active sensor connection;
- **Offline mode**, in which you may process images previously acquired and stored on your PC. This operating mode lets you simulate an Inspection using previously stored images, thus you can test or configure an Inspection without a connected sensor. It also allows focussing attention on determining the most suitable control parameters, as a constant flow of images is missing.

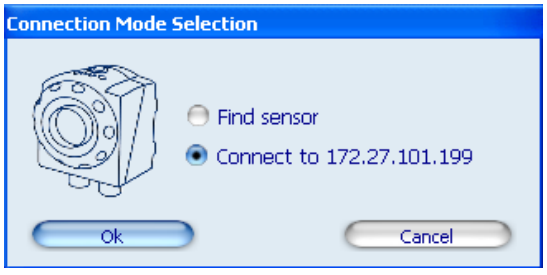
Click on the relative button to choose the desired mode.

NOTE

STEP 1 has different "work flows" in offline and online mode as described in the following sections.

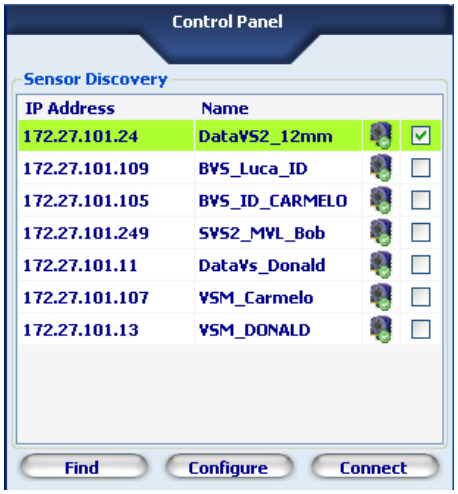
4.2.3 Establish Connection with Sensor – Online Mode Only

If you have chosen the Online mode, the software shows this window:



Parameters	Description
Find sensor	DataVS2 GUI searches for connected sensors and displays them in a list in the CONTROL PANEL.
Connect to …	DataVS2 GUI will establish a connection with the sensor addresses that correspond to the IP number displayed.

If you have selected FIND SENSOR, the software will first show a window with a progress bar. In presence of connected sensors, it will show this screen in the CONTROL PANEL at the right side of the main window:



The Control panel shows all sensors found by the Find sensor function. For each sensor the panels shows:

- the IP address of this sensors
- the Name of this sensor
- an indication whether a connection can be created (row highlighted in green colour) or not (row highlighted in red colour)
- a checkbox, representing the selected sensor for the connection. If more sensors are shown in the list you can select the sensor you want to connect by the means of the checkbox in the last column.

Once you have selected the sensor to connect, press the “Connect” button.

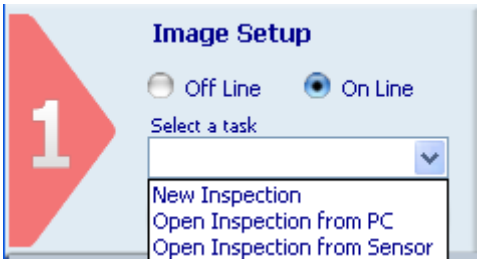
NOTE

When the sensor row is highlighted in red, the sensor cannot be reached by the DataVS2 GUI software. You may use the “Configure” button to change its IP Address and Subnet Mask, according to your current LAN Network configuration, see Chapter 3.2

4.2.4 Task Selection

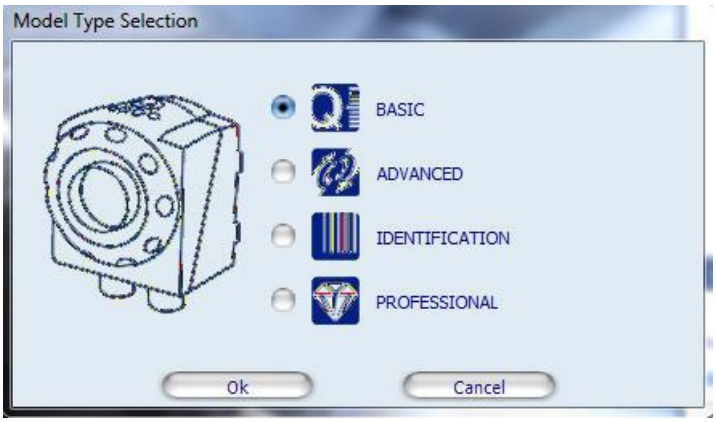
The next step for both Online and Offline modes will be selecting a task.

Click on the drop-down menu under “Select a Task” label in order to select the desired task.



Parameters	Description
New Inspection	Creates a New Inspection.
Open Inspection from PC	Loads an existing Inspection from the PC.
Online mode only: Open Inspection from sensor	Downloads an existing Inspection from the connected sensor.

If you choose “New Inspection” in Offline mode the DataVS2 GUI will first popup the “Model Type Selection” window; this will allow you to choose the sensor model you want to simulate. The DataVS2 GUI allows you to work with all available sensor model types. We suggest to always choose the same model type as your sensor one. The DataVS2 GUI will not allow to save on the connected sensor an Inspection configured for a different Model Type.



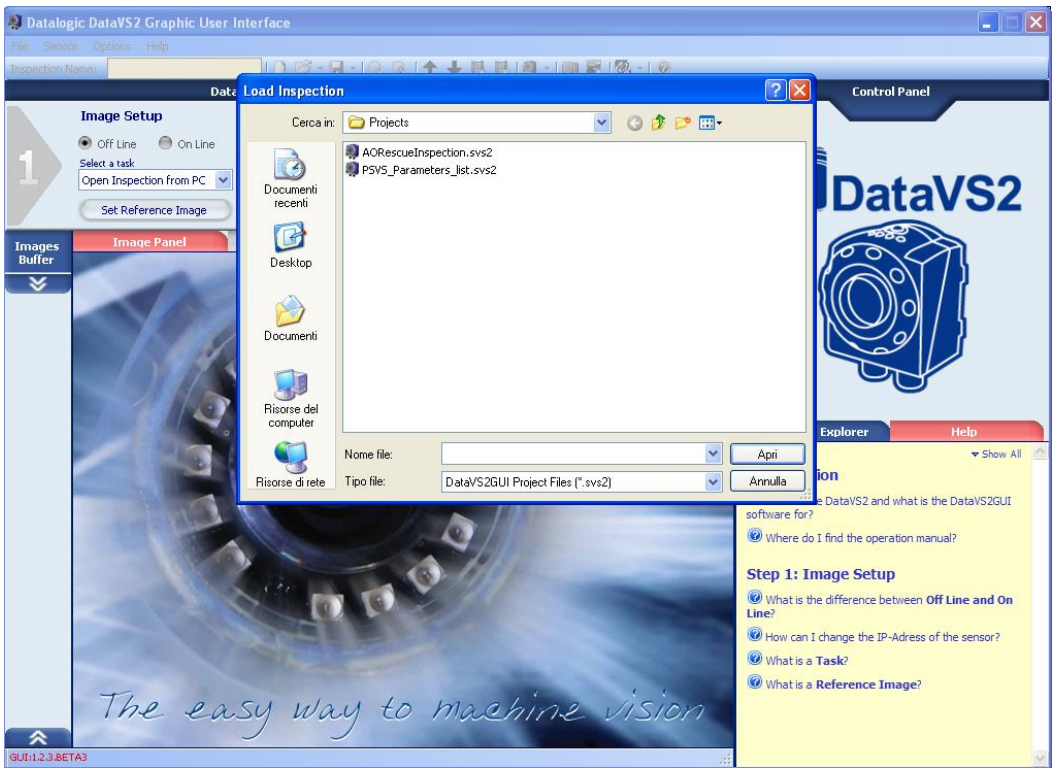
After choosing the Model Type, a File Dialog will be opened by the DataVS2 GUI: this allows you to load the images you want to integrate in your Inspection. You may load more than one image. After you have loaded these images their thumbnails are shown in the IMAGES BUFFER. You can select the current image by simply clicking on the thumbnail. You can load more images by clicking on the "Load Image" button in the CONTROL PANEL.

NOTE

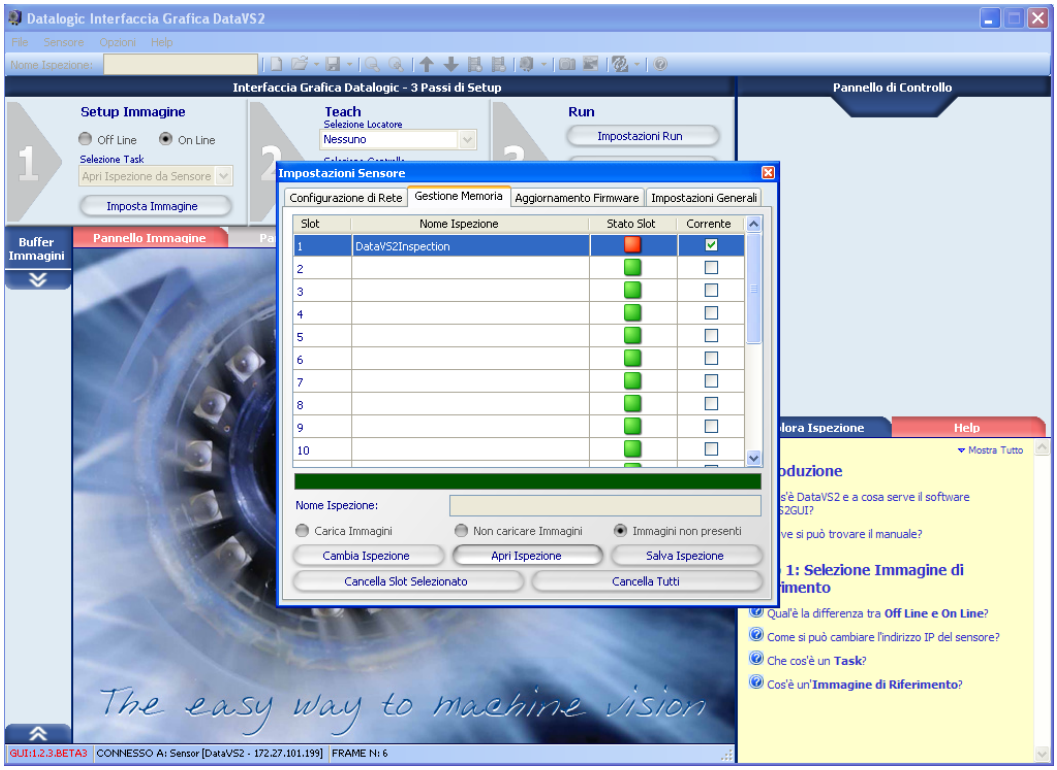
The DataVS2 GUI will allow you to work only with images previously acquired with the sensor. The first time you startup the DataVS2 GUI you will find some sample images inside the “Images” folder in the Program Installation directory.

Once you have selected your reference image, click on the "Set Reference Image" button to confirm.

If you choose "Open Inspection from PC" the DataVS2 GUI opens a File Dialog. This dialog shows the list of inspections stored on the PC. Choose one of them and press the “Open” button.

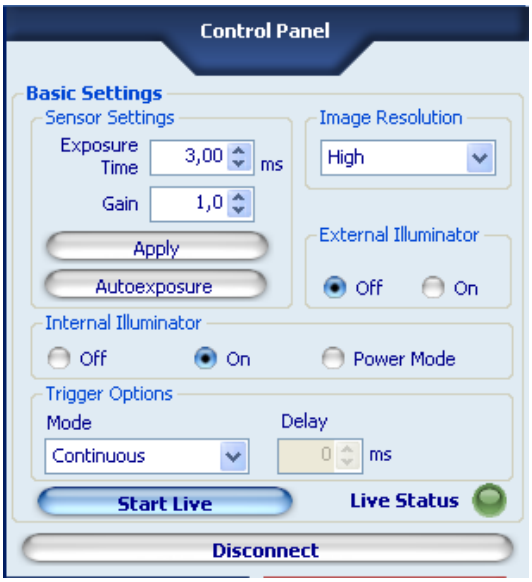


If you choose "Open Inspection from sensor" the DataVS2 GUI opens a dialog. This dialog shows the list of inspections stored on the sensor. Choose one of them and press the “Open” button.



4.2.5 Basic Settings – Online Mode Only

After you have created a "New Inspection" in the Online mode, the CONTROL PANEL will display the Basic Settings of the sensor. Before changing the Basic Settings parameters, press the “Start Live” button: in this way you will be able to see how the changes will affect images acquired by the sensor.



Parameters	Description	Default	OBJ	AOR	ID	PRO
Exposure Time	Changing the Exposure Time value will affect image brightness. The higher the Exposure time, the brighter the image acquired by the sensor.	3.0 ms	☒	☒	☒	☒
Auto exposure	Pressing the Auto exposure button, the sensor will automatically setup the best Exposure Time value depending on the current ambient light conditions.		☒	☒	☒	☒
Gain	Changing the Gain value will affect image contrast. The higher the Gain value, the greater the contrast between black and white values, and gray colours will be reduced.	1.0	☒	☒	☒	☒
Resolution	Represents the pixel resolution currently used by the sensor. This parameter affects Image dimensions in pixels. Three image resolutions are available: High: 640x480 pixels Medium: 320x240 pixels Low: 160x120 pixels The lower the resolution, the faster the image acquisition phase (and the Inspection running cycle on the sensor). Changing the Image Resolution will create automatically a New Inspection. The DataVS2 GUI software doesn’t allow you to work with different image resolutions at the same time. ***Note***: Low Image Resolution may affect the robustness of the Control execution due to the lack of image information. Avoid using Low Image Resolution when using size controls (Position, Edge Count and Width).	High	-	☒	-	☒

Parameters	Description	Default	OBJ	AOR	ID	PRO
External Illuminator	Enables or disables the external illuminator (if present) connected to PIN 3 of the M12 8-pole connector. ***Note***: The PIN 3 of the M12 connector can be used alternatively as a pulse source for the external illuminator or digital Output connected to an Inspection result. This function is available in the Hardware 3 and ST2.3.0.01 firmware (and latter versions) of DataVS2. The Digital output 4 is disabled when enabling (ON) the external illuminator.	Off	☒	☒	☒	☒
Internal Illuminator	Enables or disables the integrated illuminator. Three values are available: Off: illuminator is disabled On: illuminator is enabled and works in Normal mode Power: illuminator is enabled and works in Power mode. The light is brighter. We suggest you to lower the Exposure Time value when using the internal illuminator in Power mode, otherwise the image will be too bright.	On	☒	☒	☒	☒
Start / Stop Live	Starts or Stops Live Mode. In Live Mode, the sensor acquires images and sends them to the DataVS2 GUI.		☒	☒	☒	☒
Live	Indicates if Live mode is active (Bright Green) or not (Dark Green).		☒	☒	☒	☒

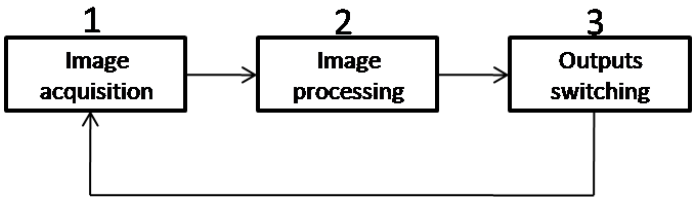
When you have selected the appropriate set of Basic parameters, continue.

4.2.6 Trigger Selection

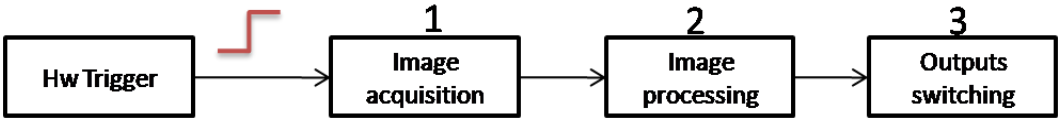
Generally speaking, the term Trigger represents a signal generated by an event that triggers another event.

In our situation, the trigger is a signal generated by a device that triggers image acquisition. When the sensor is used to inspect moving objects, it is important to ensure that the object is inside the shot area at the precise instant the image is captured for processing. For this reason, the system offers four types of triggers for the inspection configuration process:

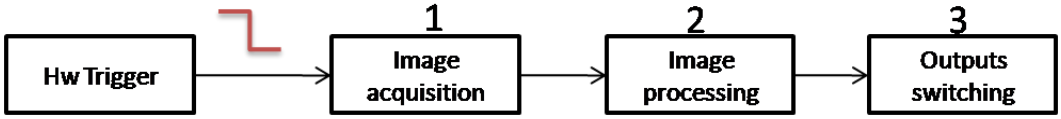
- Continuous: default trigger. System will acquire images at the fastest rate possible. A new image will be acquired as soon as the last one is processed.



- Hardware Rising: uses an external trigger signal. A new image will be acquired as soon as there is a rising edge (from 0 to 24V) at PIN 8.



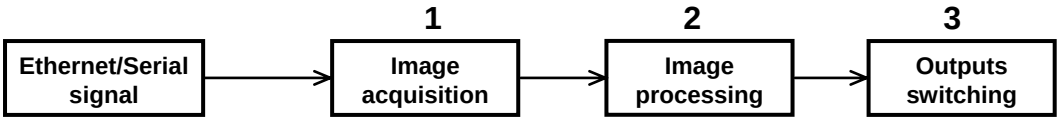
- Hardware Falling: uses an external trigger signal. A new image will be acquired as soon as there is a falling edge (from 24 to 0V) at PIN 8.



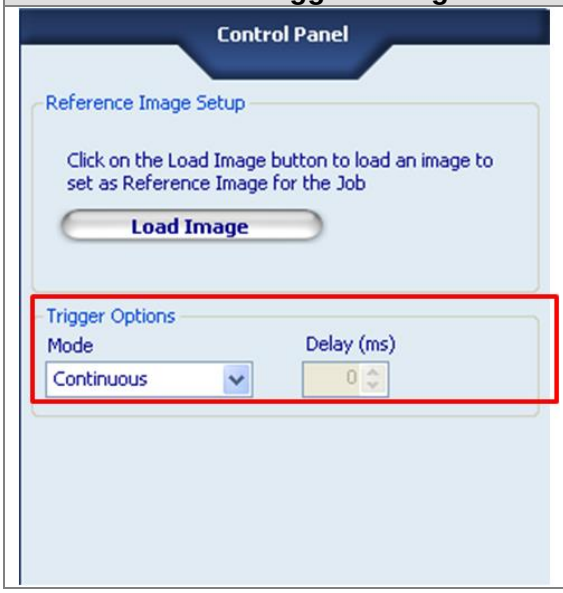
When the trigger mode is set to Hardware, it is possible to setup a special time interval, a so called **Trigger Delay**, used by the sensor to postpone the image acquisition of a well-defined time interval after the trigger event has been received.

We recommend using a Hardware trigger when the sensor is integrated in a machine environment. It is absolutely necessary to use a Hardware trigger to inspect any moving object; it is important to ensure that the object is inside the field of view of the sensor when the sensor acquires the image.

- Ethernet/Serial: uses a character string as a trigger via RS232 Serial or Ethernet communication. A new image will be acquired as soon as the signal is received.



Select the type of Trigger used to define Inspection time instants; different panels are shown for Trigger configuration depending on the current operation mode, Online or Offline, as shown in the table below.

Offline mode Trigger configuration	Online mode Trigger configuration
	

The following table contains a brief description of the parameters used to configure the Trigger options.

Parameters	Description	Default
Trigger setting: Mode	Represents the type of trigger used to define Inspection time instants. Four Trigger Modes are available: • Continuous: default trigger. Sensor will acquire images at the fastest speed possible. A new image will be acquired as soon as the last one has been processed. • Hardware Rising: uses an external hardware trigger e.g. a sensor. A new image will be acquired as soon as there is a rising edge (0 to 24 V) on PIN 8. • Hardware Falling: uses an external hardware trigger e.g. a sensor. A new image will be acquired as soon as there is a falling edge (24 V to 0 V) on PIN 8. • Ethernet / Serial: trigger signal is composed of a message sent to the sensor through the Serial interface or Ethernet. The message is represented by a string of “TRIGGER” characters ending with NULL. A new image will be acquired as soon as the message is received by the sensor. ***Note***: the Sensor→Setting panel allows to set a Trigger different from the one used by DataVS2 GUI. ***Note***: this function is available in ST2.3.0.01 firmware, and later versions, of DataVS2.	Continuous
Trigger setting: Delay	Represents the time interval (in milliseconds) between the instant when the external trigger signal occurs and the next image is acquired by the sensor. Trigger Delay control is active and selectable only when Trigger Mode is set on “Continuous”.	Disabled
Trigger setting: Trigger button	Allows to send an Ethernet or Serial type of Trigger to the sensor. An image is acquired by the connected sensor every time the button is pressed. The button can be seen only when the Trigger mode parameter is set on the “Ethernet / Serial” value. When visible, the button is enabled only after pressing “Start Live”. ***Note***: this function is available in ST2.3.0.01 firmware, and later versions, of DataVS2.	Disabled

Once you have selected the desired type of trigger, you may test it by starting the LIVE MODE and verify the images acquired by the sensor.

4.2.7 Reference Image Setup

Finally you have to select the reference image. By clicking on the “Set Reference Image” button you choose the "Master Image" which will be the base of your Inspection.

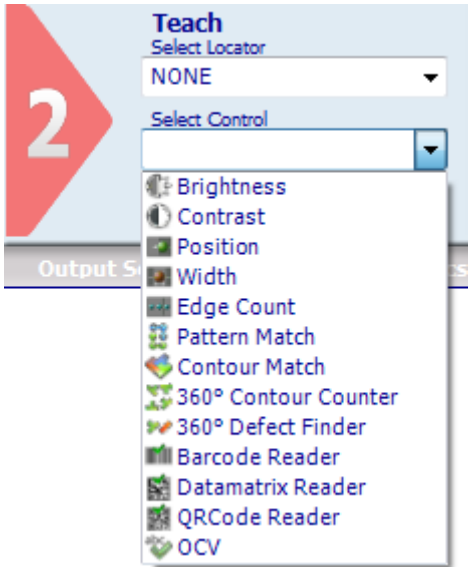
4.3 STEP 2: TEACH

After setting your Reference Image you may now proceed with STEP 2. The following operations are available in this STEP:

- Insert a Locator Tool to allow the sensor to find the object inside the image
- Insert Control Tool to allow the sensor to check the object characteristics
- Output configuration.

4.3.1 Quick Guide STEP 2

1. Click on the drop-down menu under the “Select Control” label and choose a Control from the list shown according to the inspection needed.

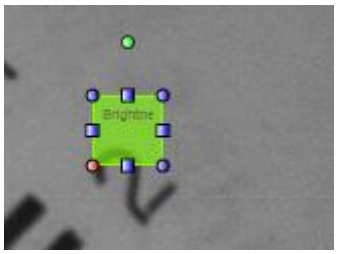


NOTE

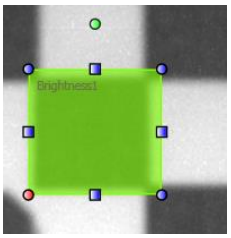
The Controls pool changes depending on the currently Selected Sensor Model. For a complete list of controls per model, refer to the table in par. 5.2.2.

2. Once a Control has been selected, drag the Control icon over the image to the position where inspection is to be performed.
3. Click on the Control icon (mouse pointer) to anchor it in that position.

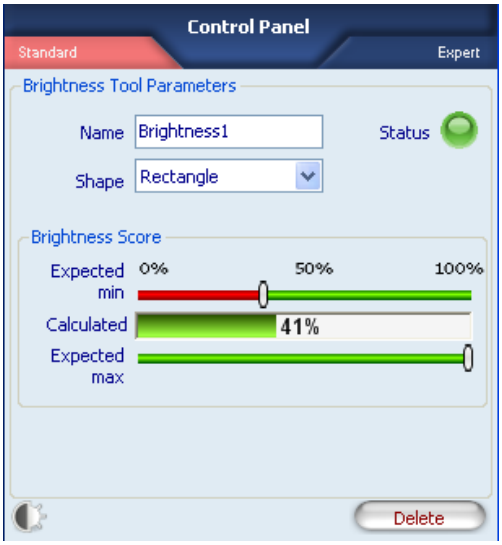
Now you can see a delimited "Region of Interest" (ROI) (see picture below) inside the image:

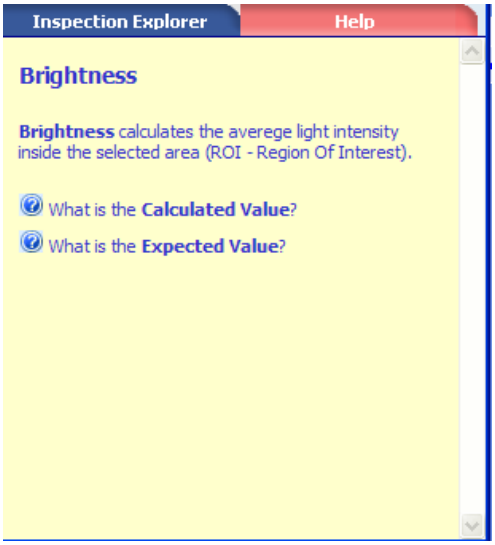


4. Place the cursor over any of the 4 ROI borders, a 4 arrow head cursor should appear. Click and hold while dragging the Region of Interest frame to adjust the overall position in the image.
5. Place the cursor over any of the 8 ROI anchors; click and hold the mouse left button while dragging the anchor. Now resize the frame so that the object area to be inspected is fully covered, as shown below:

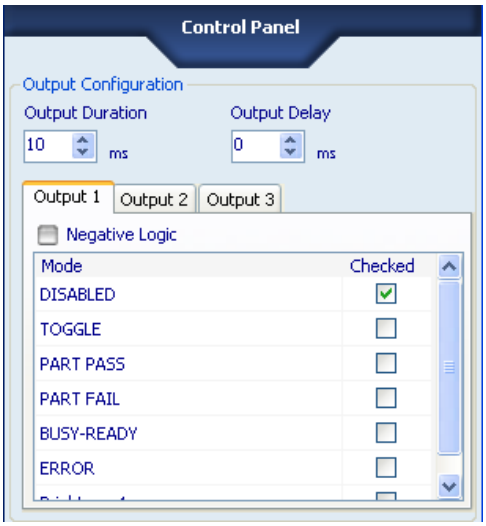


6. The DataVS2 GUI software will now display the parameters of the current Control, at the right side of the screen in the CONTROL PANEL. Change these parameters as necessary to create the most reliable Control performance. The online help on lower right side of the screen provides information on each parameter and how to use these tools.





7. If you want to insert more Controls you have to start with "Select Control" again.
8. Once all the Controls have been created and set, click on the "Output Setup" button to configure the sensor's outputs. The DataVS2 GUI software will now display the "Output Configuration" settings for the three available outputs.



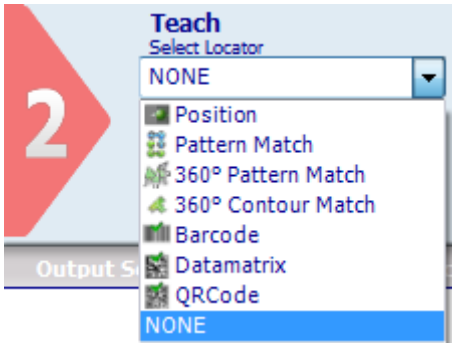
9. For each of the three Output Tabs in the CONTROL PANEL choose the appropriate "Output Mode" configuration by selecting the checkbox corresponding to the desired value. Above Output Mode tables, the overall "Output duration" and "Output delay" can be setup.
10. Once all the outputs have been created and setup, click on the "Test" button directly under the "Run Settings" button and proceed with STEP 3.

4.3.2 Locator Selection

A Locator is a special Tool which searches inside the whole image for a certain characteristic e.g. a gray value edge or a pattern and determines its position. Once the position has been calculated it is used as the relative reference system for all other Controls included in the same Inspection. If the locator has not found any position in the current image, all other Controls in the Inspection will not be calculated and their result will be FAILURE.

NOTE *If you move your mouse cursor in the IMAGE PANEL after you have selected a locator from the pull-down menu under the "Select Locator" label, it will change appearance; it will show the icon of the Locator Tool as shown in the menu.*

NOTE *Maximum one locator in each Inspection.*



Parameters	Description	OBJ	AOR	ID	PRO
None	No Locator is used in the Inspection.	☒	☒	☒	☒
Position	Locates the position of an edge in pixels. The edge is found searching along a direction passing through the ROI center.	☒	☒	☒	☒
Pattern Match	Searches for a pattern inside the ROI; the pattern template is acquired in the Teach phase (STEP 2). The pattern to be found must have the same orientation of the template.	☒	☒	☒	☒
360° Pattern Match	Searches for a pattern inside the ROI; the pattern template is acquired in the Teach phase (STEP 2). The pattern to be found may have a different orientation with respect to the template first pattern.	-	☒	-	☒
360° Contour Match	Searches for a contour of an object inside the ROI; the contour is acquired in the Teach phase (STEP 2). The object to be found may have a different orientation with respect to the template.	-	-	-	☒
Datamatrix	Searches for, locates and decodes a Datamatrix code present inside the ROI	-	-	-	☒
Barcode	Searches for, locates and decodes a Barcode code present inside the ROI	-	-	-	☒
QRCode	Searches for, locates and decodes a QR code present inside the ROI	-	-	-	☒

NOTE

After inserting a Locator, the other Controls may not be visible in the IMAGE PANEL and DataVS2 GUI only displays the Locator ROI (with red borders). This means that the Locator Tool fails. Please, change the position of the Locator ROI or adjust the Locator parameters in order to allow the Locator find a good target. As soon as the Locator is OK again, all other control ROIs will be displayed.

4.3.3 Control Selection

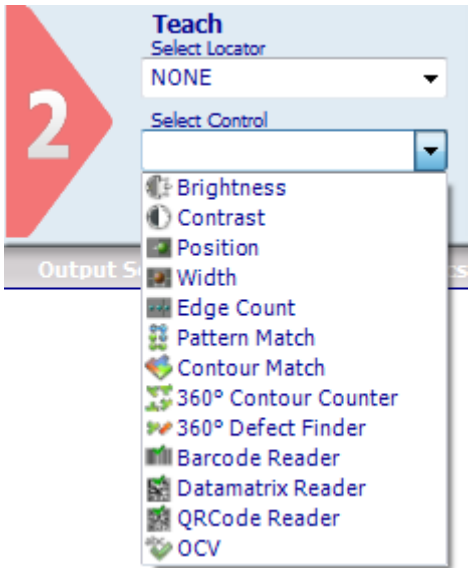
Select the Control you want to add to the current Inspection using the pull-down menu. If you want to add more than one Control to your Inspection you are allowed to do so up to a maximum of 255 Controls.

NOTE

If you move your mouse cursor in the IMAGE PANEL after you have selected a Locator from the pull-down menu under the "Select Control" label, it will change appearance; it will show the icon of the Control Tool as shown in the menu.

NOTE

Please consider that the more the Controls an Inspection has, the higher the processing time for the Inspection.



Parameters	Description	OBJ	AOR	ID	PRO
Brightness	The average intensity of pixels inside the ROI.	☒	☒	-	☒
Contrast	Calculates the ratio between the brightest and the darkest pixels inside the ROI.	☒	☒	-	☒
Contour Match	Searches for a specific shape inside the ROI; the shape template is acquired in the Teach phase (STEP 2)	☒	☒	-	☒
Edge Count	Counts the number of edges inside the ROI searching along a direction passing through the ROI center.	☒	☒	-	☒
Width	Measures the distance between two edges in pixels.	☒	☒	-	☒
Pattern Match	Searches for a pattern inside the ROI; the pattern template is acquired in the Teach phase (STEP 2). The pattern to be found must have the same orientation as the template.	☒	☒	-	☒
Position	Locates the position of an edge in pixels. The edge is found searching along a direction passing through the ROI center.	☒	☒	-	☒
Datamatrix Reader	Searches and decodes one or more Datamatrix symbols in a ROI.	-	-	☒	☒
Barcode reader	Searches and decodes one or more Barcode symbols in a ROI.	-	-	☒	☒
OCV	Verifies the presence and print quality of a character string.	-	-	☒	☒
360° Contour Counter	Searches and counts a group of objects inside a ROI; the objects are searched and identified by analysing the contours	-	-	-	☒
360° Defect Finder	Detects the defect level of an object with respect to the template present in the reference image.	-	-	-	☒
QRCode Reader	Searches and decodes one or more QR symbols in a ROI.	-	-	☒	☒

Once selected, you have to position your Control in the reference image. This is done graphically by positioning and sizing its ROI inside the working area. After you have selected a Control, the CONTROL PANEL displays the default parameters for this Control. To adapt the parameters, use the graphical controls provided such as sliders and checkboxes.

The calculated result of a Control is indicated by the Status indicator turning green or red; the frame of the ROI shown inside the IMAGE PANEL will be displayed in the same colour as the status indicator.

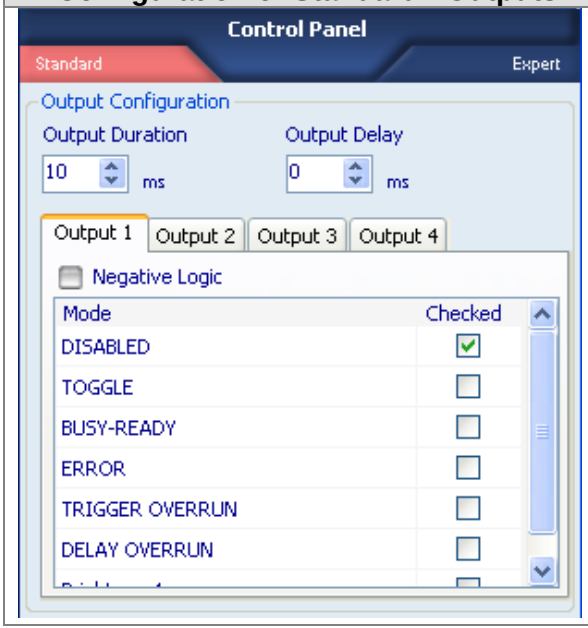
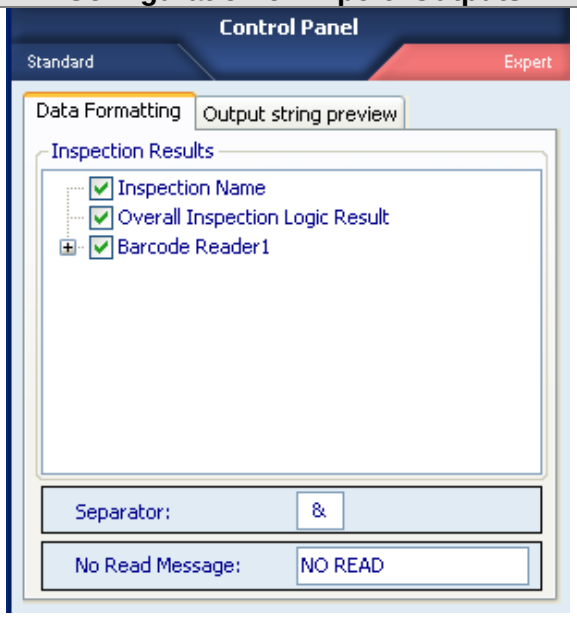
4.3.4 Configuration of Outputs and Image Saving Condition

Once you click on the “Output Setup” button, the CONTROL PANEL allows you to set up the sensor outputs and the image saving condition of an Inspection currently in execution.
You can setup the desired output configuration for all the three digital outputs: by default they are disabled.

The same configuration steps can be repeated to set the image saving condition. If enabled the acquired images will be saved according to a FIFO (First In First Out) policy. The sensor can store up to 10 images.

NOTE

You can download and see the images saved on sensor by opening the currently running Inspection on the sensor. If present, the images saved will be downloaded from the sensor and shown in the IMAGE BUFFER.

Configuration of Standard - Outputs	Configuration of Expert- Outputs
	

Parameters	Description	Default	OBJ	AOR	ID	PRO
Mode	Defines the output configuration associating a certain logical result to this output. ***Note***: DataVS2 offers 4 digital outputs that can be associated to Inspection results. This function is available in the Hardware 3 and ST2.3.0.01 firmware (and latter versions) of DataVS2. PIN 3 of the M12 connector can be used alternatively as a pulse source of an external illuminator or digital output 4 connected to an Inspection result. The Digital output 4 is disabled when enabling (ON) the external illuminator.	Disabled	☒	☒	☒	☒
Negative logic	If you select the Negative Logic, the output will be low when the Control result is SUCCESS. In default settings, the output will be high when the Control result is SUCCESS.	Disabled	☒	☒	☒	☒
Output Duration	Determines the output duration in milliseconds. The output will maintain its last value even after the timeout defined in “Output Duration”. Default output duration is 10 ms. With this configuration, all outputs will maintain their values for 10 ms after an Inspection result is available. When the Output duration is 0 ms, the outputs maintain their status until the next result is available. ***Note:*** changing Output Duration will affect all three outputs.	10 ms	☒	☒	☒	☒
Output Delay	Determines how much time the output has to wait before displaying the last acquired result. Output Delay is calculated from the end of the image acquisition. Default delay is 0 ms and so the outputs are switching immediately after an inspection result is available. If the Inspection execution time exceeds the delay interval, the sensor goes in ERROR signalling: you can see the ERROR message by setting one of the Output Modes to ERROR. This kind of error doesn’t block sensor inspection. ***Note:*** when testing an Inspection in the RUN mode with Statistics enabled on the DataVS2 GUI graphic interface, the Output Delay is displayed in the STATISTICS Panel. ***Note:*** changing Output Delay will affect all outputs.	0 ms	☒	☒	☒	☒

Parameters	Description	Default	OBJ	AOR	ID	PRO
Result format	Allows to compose the message containing the inspection results sent to the sensor at the end of each inspection on the Ethernet or Serial channel. The message containing the results is composed of all the elements selected in the “Result Format” area. Please refer to Chapter 8 for more information on the protocol to use for results reading. ***Note***: this function is available in ST2.3.0.01 firmware, and latter versions, of DataVS2. ***Note***: the string result preview can be visualized by selecting the Result Preview Tab.	The default setting of DataVS2 returns the complete string results.	-	☒	☒	☒
Separator	Represents the character inserted as a separator between one result and the successive one inside the message containing inspection results. The separator value can be changed by simply inserting the desired value in the specific area. ***Note***: this function is available in ST2.3.0.01 firmware, and latter versions, of DataVS2.	‘&’	-	☒	☒	☒
NO READ message	Represents the character string included in the result message instead of each Barcode, Datamatrix or QRCode tool that has generated a failure result. When the NO READ string is not configured, the sensor returns the normal set of results, obtained amongst those marked as necessary in the result Format panel. When the NO READ message is configured, a NO READ string is returned instead of each Datamatrix and Barcode tool having a FAILURE result. ***Note***: this function has been introduced beginning from the ST2.4.0.XX firmware release and is supported only by the Inspections having the 4.0 version value.		-	-	☒	☒

NOTE

Ensure that the sum of Delay Time and Duration Time is always considerably less than the time period between two parts to be inspected. If you infringe this rule, the sensor is not able to give a result for each part.

The following table contains an ordered description of all available Output Modes.

Parameters	Description	OBJ	AOR	ID	PRO
DISABLED	The Output is disabled; the digital value is always LOW (or HIGH with Negative Logic flag checked).	☒	☒	☒	☒
PART DETECTED	The Output is associated to the result of the Locator (if present). If the result of the Locator is SUCCESS, the output is HIGH, otherwise the output is LOW.	☒	-	-	-

Parameters	Description	OBJ	AOR	ID	PRO
	Note: if the Inspection doesn't contain any Locators, this Output Mode value is hidden. ***Note***: this output mode is not present in the DataVS2 AOR model. The Logical Tools can be used to obtain this function.				
PART PASS	The Output is associated to the logical NAND of the Controls currently present in the Inspection. If the result of each of the Controls is SUCCESS, the output is HIGH, otherwise the output is LOW. Note: this output mode is not present in the DataVS2 AOR model. The Logical Tools can be used to obtain this function.	☒	-	-	-
PASS FAIL	The output is associated to the logical NAND of the Controls currently present in the Inspection. Even if only one of the results is FAIL, the output is HIGH, otherwise the output is LOW. Note: this output mode is not present in the DataVS2 AOR model. The Logical Tools can be used to obtain this function.	☒	-	-	-
CONTROL RESULT	The output is associated to the logical result of the Control corresponding to the selected row. If the Control result is SUCCESS the output status is HIGH, otherwise the output status is LOW.		☒	☒	☒
LOGICAL TOOL RESULT	The output is associated to the logical result of the Logical Tool corresponding to the selected row. If the Control result is SUCCESS, the output status is HIGH, otherwise the output status is LOW.		☒	☒	☒
BUSY-READY	In this configuration the output maintains the HIGH value for the entire time interval corresponding to the Inspection execution time; as soon as the result is available and the Inspection execution cycle is completed, the output value becomes LOW. ***Note***: this output mode is not displayed by the DataVS2 GUI when in the RUN mode.	☒	☒	☒	☒
ERROR	In this configuration, the output becomes HIGH when an ERROR occurred on the sensor; three are the main ERROR situations signalled by the sensor: A software crash occurs during Inspection execution: in this situation the sensor is blocked and has to be reset. An output delay time out: in this situation the execution time has exceeded the delay time set by the user; the sensor is not blocked after an output delay time out. A trigger delay error: in this situation the sensor has received a trigger input for the next image acquisition when the previous execution cycle is not completed; the sensor is not blocked after a trigger delay error. ***Note***: this output mode is not displayed by the DataVS2 GUI when in the RUN mode.	☒	☒	☒	☒
TOGGLE	In this configuration the output switches after every sensor Inspection. There is no relationship with the Inspection result. After every inspection execution, the status of the output configured as TOGGLE switches from 0 to 1 and vice versa. ***Note***: this output mode is not displayed by the DataVS2 GUI when in the RUN mode.	☒	☒	☒	☒

The following table displays the possible Images Saving conditions:

Parameters	Description	OBJ	AOR	ID	PRO
ALL	If selected, every Inspected image will be saved on the sensor.	☒	-	-	-
ON PART DETECT	If selected, only images corresponding to a Locator Success will be saved on the sensor. ***Note***: this option is shown only if the current Inspection contains a Locator.	☒	-	-	-
ON PART PASS	If selected, only images corresponding to an Inspection Success (result of all the Controls is SUCCESS) will be saved on the sensor.	☒	-	-	-
ON PART FAIL	If selected, only images corresponding to an Inspection Failure will be saved on the sensor.	☒	-	-	-
CONTROL RESULT	If the control result is successful the images will be saved to the sensor	-	☒	☒	☒
LOGICAL TOOL RESULT	If the logical tool result is successful the images will be saved to the sensor	-	☒	☒	☒

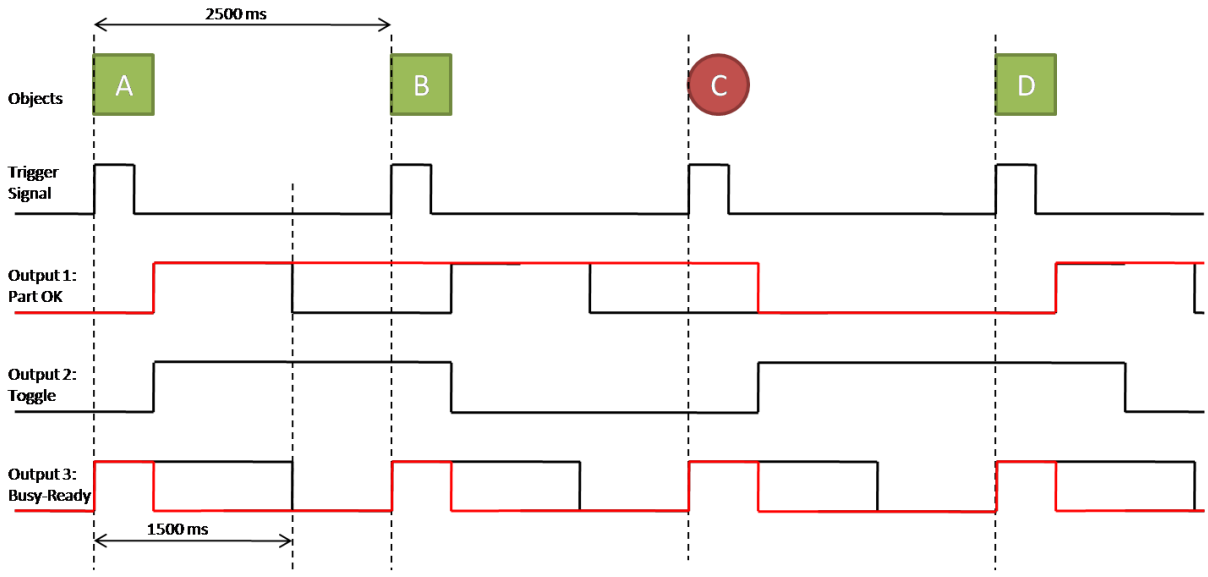
4.3.5 Example of a Timing Diagram

The following diagram displays an example of Output timing and signals.

Let's assume to test 4 parts (A, B, C, D) moving on a conveyor belt: part C is not OK. Every 2500 milliseconds (ms) a new part has to be inspected. We assume an execution time of 500 ms. Output Delay is set to 0 ms. The Output Duration is set to 1000 ms.

Output Modes are connected according to the following description:

- Output 1: part OK
- Output 2: Toggle
- Output 3: Busy-Ready



The explanation of the timing diagram displayed in the above picture is provide below:

- The sequence for the trigger signal (at the top of the diagram) shows the trigger signal received by the sensor; in this example we assume the inspection execution starts with the rising flank of the trigger (Trigger Mode is set to Hardware Rising).
- The Busy-Ready Output goes HIGH when a Trigger input is received. The Busy-ready Output changes to LOW when the Inspection is complete. The black sequence

shows the behaviour of this output when Output Duration is set to 1000 ms. The red sequence represents the behaviour of this output when Output Duration is set to 0 ms; in this case the Busy-Ready signal is set to LOW immediately after an Inspection result is available.

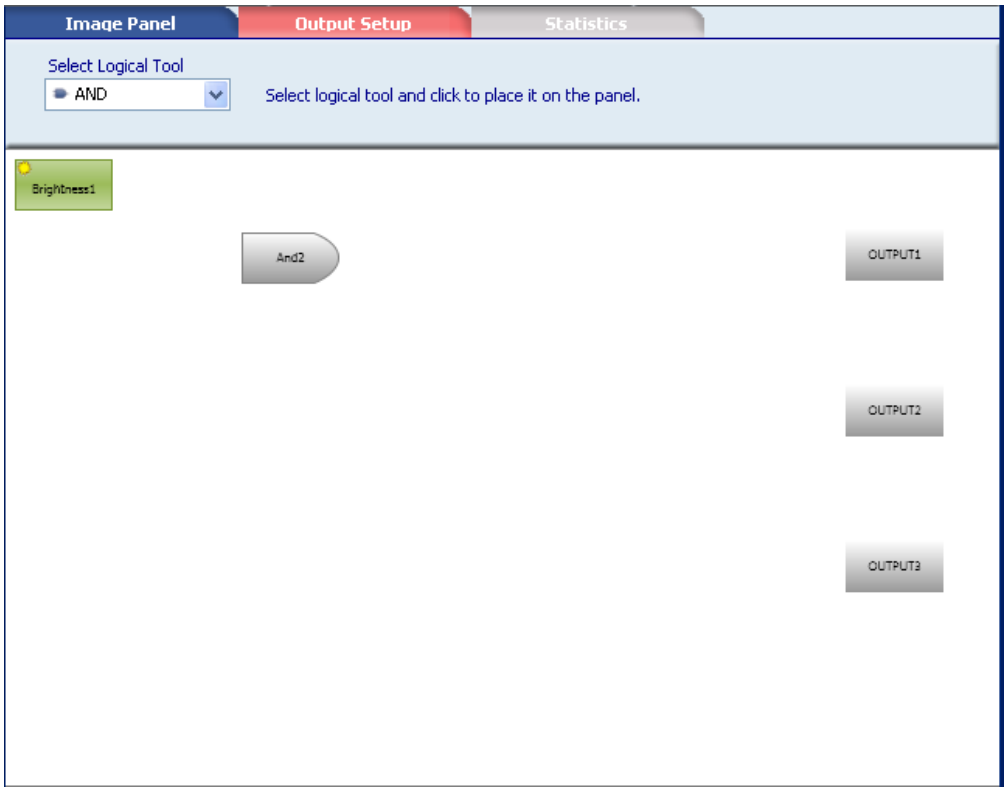
- Toggle Output is set to HIGH as soon as the Inspection result for part A is available. After the availability of the result for part B, the Toggle signal changes to LOW. Every time the Toggle signal changes its status a new Inspection is available. This distinguishes two consecutive inspection results with the same status (e.g. when two adjacent parts are wrong)
- Output 1 is set to part OK. The black sequence shows the behaviour of this output when Output Duration is set to 1000 ms; in this case the output switches to HIGH as soon as a good part is inspected. After staying HIGH for the set Output Duration the status changes back to LOW again. In the diagram above there is no status change for part C, as it is a not good part; the output stays LOW until the result of the next good part is available.
- The red sequence shows the behaviour of this output when Output Duration is set to 0 ms. In this case the output switches to HIGH after part A, a good part, is inspected and does not fall back to LOW until a bad part is found (part C in the diagram above). After going LOW the output stays LOW until a new good part is inspected.

4.3.6 Logical Tool Selection

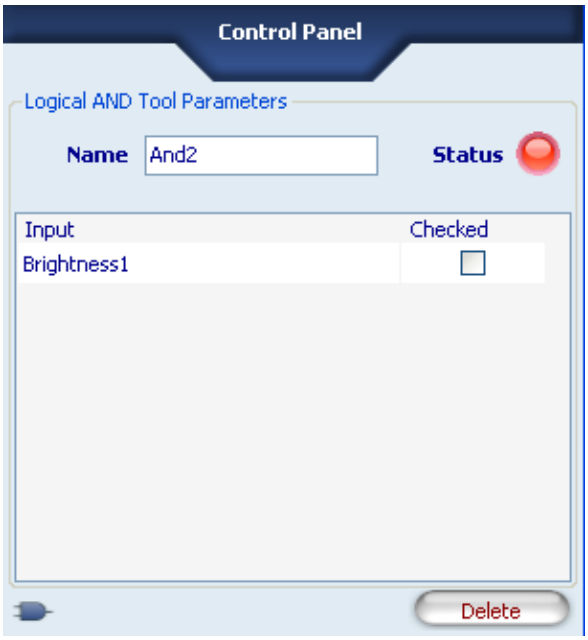
NOTE

This feature is not enabled only when the current Inspection is a DataVS2 OBJ model type; you can always check the Model Type value of the current inspection from the INSPECTION EXPLORER panel, see Chapter 3.

When you click on the “Output Setup” button, the OUTPUT SETUP TABLE will be displayed in the central area of the main window, allowing you to insert and configure Logical Tools.
Opening the LOGICAL TOOL TABLE you will see the list of the Controls currently inserted in the Inspection on the left side and the three digital Outputs on the right, both shown as rectangle boxes. Controls boxes are highlighted in red or green depending on the logical result of the Control, failure or success. Output boxes are highlighted in gray or yellow depending on the logical status of the output, low or high.
Select the Logical Tool you want to add to the current Inspection by the means of the pull-down menu. Once selected you have to position your Logical Tool in the Logical Tool container. This is done graphically by left-clicking the mouse in the position you want to locate the Logical Tool. An image will then appear representing the logical gate corresponding to the inserted Logical Tool.

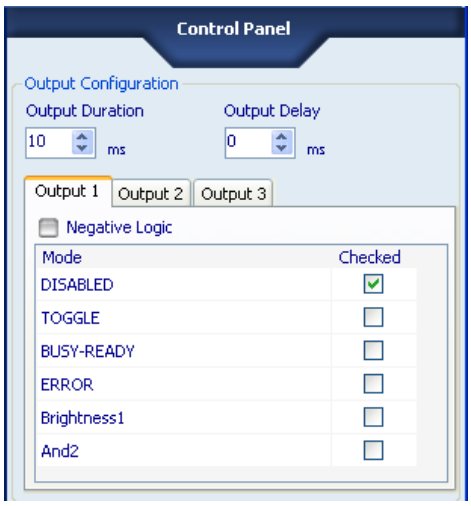


After selecting the Logical Tool, the CONTROL PANEL will display the Logical Tool parameters, represented by the list of all possible Controls that can be connected as inputs to the Logical Tool; the currently connected inputs are shown as checked inputs in the CONTROL PANEL. As soon as the minimum number of Inputs are connected to the Logical Tool, it will change colour and become green (SUCCESS) or red (FAILURE) depending on the logical status of the gate.



Parameters	Description	OBJ	AOR	ID	PRO
AND	Calculates the logical AND of the Controls connected as inputs.	-	☒	☒	☒
OR	Calculates the logical OR of the Controls connected as inputs.	-	☒	☒	☒
XOR	Calculates the logical XOR of the Controls connected as inputs.	-	☒	☒	☒
NAND	Calculates the logical NAND of the Controls connected as inputs.	-	☒	☒	☒
NOR	Calculates the logical NOR of the Controls connected as inputs.	-	☒	☒	☒
XNOR	Calculates the logical XNOR of the Controls connected as inputs.	-	☒	☒	☒
NOT	Calculates the logical NOT of the Controls connected as inputs.	-	☒	☒	☒
MAJORITY	Calculates the minimum number of input Controls whose result must be SUCCESS in order to make the Logical Tool result SUCCESS too.	-	☒	☒	☒

Once connected the Logical Tool to its inputs you have to connect it to Outputs; to do so, from the Logical Tool container select the output mode you want to connect the Logical Tool to, and from the CONTROL PANEL select the Logical Tool row as output Mode.



NOTE

The maximum number of Inputs that can be connected to a Logical Tool depends on the Logical Tool type; please, refer to Chapter 0 for a complete explanation of Logical Tools.

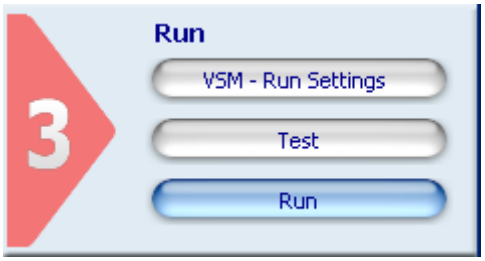
NOTE

More than one Logical Tool can be added to your Inspection. The maximum number of Tools (Controls, Locators and Logical Tools) that can be inserted in an Inspection is 255.

4.4 STEP 3: RUN

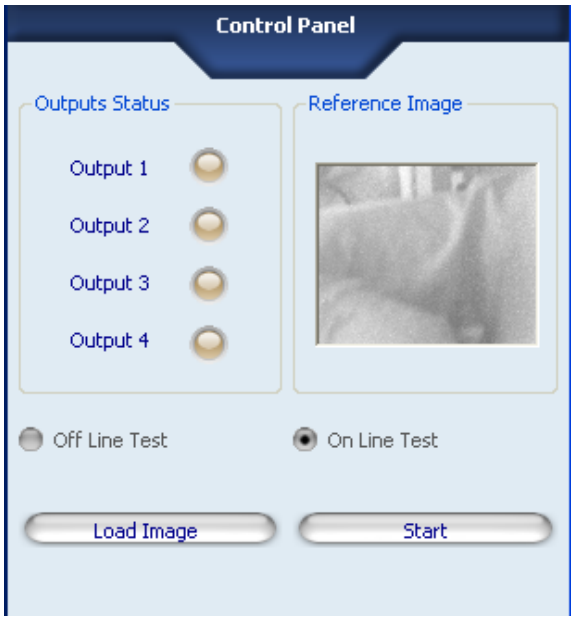
If you have successfully inserted all Controls you need and configured Outputs, proceed with STEP 3. In this STEP you will be able to:

- Configure Run Settings for the current Inspection
- Test your Inspection either Offline or Online
- Run your Inspection if Online



4.4.1 Quick Guide STEP 3

1. Click on the “Run Settings” button in STEP 3. The DataVS2 GUI software will display some running options in the CONTROL PANEL.
   - The “Images Saving” settings can be used to save Inspection related images on the sensor during its execution. The images will be saved in a queue according to a FIFO (First In First Out) policy.
   - The “Teach" button setting allow you to enable or disable the teaching behaviour of the button integrated on the sensor, see Chapter 7.1.
2. Once the Run Settings have been configured click on the “Test” button under the “Run Settings” in STEP 3.
   The DataVS2 GUI will display the “Output Status” and the “Reference Image” configured for the Inspection.



If you are in Offline Mode you can only make an Offline Test: to do so click on the "Load Image" button or click on the IMAGE BUFFER icon in the Tool Bar and select up to 19 images to cycle through for testing.

NOTE

The IMAGE BUFFER stores images into a circular array, according a FIFO policy. After the number of loaded images has reached the maximum array dimension, the next loaded image will overwrite the older one. If you need to load more than 19 images at the same time (and enlarge the array dimension) you can change the IMAGES BUFFER dimension from the OPTIONS Menu.

If you are in Online Mode you can perform an Online Test, by clicking on the "Start" button. The DataVS2 GUI software will start loading LIVE images and testing each according to the Inspection settings.

NOTE

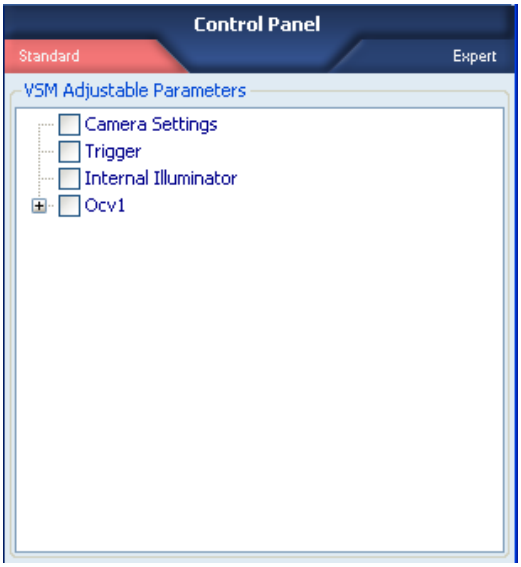
For both Online and Offline Test, the "Output Status" LED shown in the CONTROL PANEL will change according to the settings made for each output.

NOTE

Sensor Outputs are physically disabled in Test mode.

3. To stop an Online test, click on the "Stop" button.
4. To run the Inspection, click on the "Run" button directly under the "Test" button in STEP 3. The CONTROL PANEL will display the Run Settings controls.
5. Choose the "Statistics and Timings" option for the run mode.
6. Now press "Start" button"; if the Inspection has not been saved on the sensor, a dialog box will appear asking you to save the Inspection in one of the available Inspection Slots on the sensor. Each Inspection can be saved with an individual name for easier identification.

4.4.2 Run Settings



After clicking on the “Run Settings” button in STEP 3, the CONTROL PANEL will display the following options:

Parameters	Description	Default
VSM Modifiable Parameters	Selects the parameters of the running inspection that have to be displayed and changed using the VSM when connected to DataVS2. A parameter can be displayed by selecting the corresponding box. ***Note***: this function is available in ST2.3.0.01 firmware, and latter versions, of DataVS2.	None

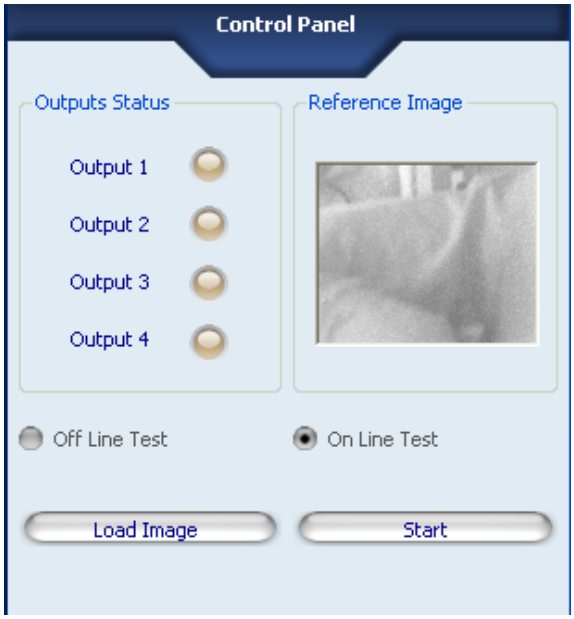
4.4.3 Test

After clicking on the “Test” button available in STEP 3, the CONTROL PANEL shows:

- the simulated status of each output (yellow for HIGH, brown for LOW);
- the currently used Test mode (either Offline test or Online test);
- the Reference Image used for the Inspection.

NOTE

Sensor Outputs are disabled in the Test mode. In the Online Test, the images are acquired by the sensor. Results are calculated by the DataVS2 GUI in both Test modes, Offline and Online.



Button	Description
Add Image	Offline Mode only: it allows you to add new images to the IMAGE BUFFER which shall be tested according to the current Inspection settings.
Start/ Stop Test	Online Mode only: sensor will start to acquire images; the Inspection is performed by the DataVS2 GUI graphic interface on the PC; results are updated;
Trigger	Online Mode only: when an Ethernet / Serial Trigger mode is set, the Trigger button triggers the inspection from the sensor and acquires an image on PC; at the end of inspection the results are displayed on the DataVS2. ***Note***: the button is enabled by DataVS2 GUI only after having pressed “Start Test”. ***Note***: this function is available in ST2.3.0.01 firmware, and latter versions, of DataVS2.

In Offline Mode, an image is tested after you have selected it from the IMAGES BUFFER. The results are displayed in the IMAGE PANEL and the Inspection Explorer is updated.

4.4.4 Run

Run Mode is accessible only in the Online Mode. If you click on the “Run” button in STEP 3, the CONTROL PANEL will show:

- the current status of each output (yellow for HIGH, brown for LOW)
- the Reference Image used for the current Inspection.

Parameters	Description
Graphics Enabled	The currently processed image is sent to the PC and shown in the IMAGE PANEL.
Statistics and Timings	The sensor does not send images to the PC. The DataVS2 GUI graphic interface shows only the inspection results for each Control, the timing for the current Inspection phases and the statistics for the current Inspection. These values may be used to evaluate the performance of the Inspection currently performed by the sensor.
Start / Stop	After pressing the “Start” button, the DataVS2 GUI graphic interface will ask you to save the current Inspection in one of the 20 sensor Slots available. A different name can be assigned to each Inspection to ease identification. ***Note***: If your sensor is running with a firmware version older than ST.2.1, you will see only 8 memory Slots available for Inspection saving.
Trigger	When an Ethernet / Serial Trigger mode is set, the Trigger button triggers the inspection from the sensor and acquires an image on PC; at the end of inspection the results are displayed on the DataVS2. ***Note***: the button is enabled by DataVS2 GUI only after having pressed “Start”. ***Note***: this function is available in ST2.3.0.01 firmware, and latter versions, of DataVS2.

NOTE

Run mode is not accessible if the current sensor Model Type does not correspond to the current Inspection Model Type. In this case you will see the “Run” button disabled in STEP 3. In order to run your Inspection on the sensor you have to select the sensor’s Model Type from the Tool Bar and configure again a New Inspection.

The sensor has now been configured and is ready for operating in a stand-alone mode.
If the Inspection performance fulfil your needs you can now disconnect the sensor from the DataVS2 GUI software. Press the pull-down menu under the “Sensor” Menu and select “Disconnect from sensor”. The sensor will start running as a stand-alone device. You should also disconnect the Ethernet cable from the sensor.

4.5 INSPECTION SETTINGS

The following table lists all the Inspection parameters that you can configure through the DataVS2 GUI software.
The Default Value represents for each parameter the default value that the DataVS2 GUI software will set at Inspection creation.
The values column contains all the admitted values for the corresponding parameter.
The DataVS2 OBJ, AOR and ID/PRO columns refer to the Inspection Model Type and are checked depending on the availability of the parameter in the selected Model.

Parameters	Default values	Values	OBJ	AOR	ID / PRO
Version	5.0	1.2 2.1 3.0 4.0 5.0	☒	☒	☒
Model Type		OBJ AOR ID PRO	☒	☒	☒
Name	SvsInspection		☒	☒	☒
Image Resolution	640x480 pixel	640x480 pix 320x240 pix 160x120 pix	-	☒	☒ (*)
Exposure Time	3 ms	[0,1 - 100] ms	☒	☒	☒
Gain	1	[1- 3]	☒	☒	☒
Trigger Mode	Continuous	Continuous Hardware Ethernet / Serial	☒	☒	☒
Hardware Trigger Polarity	Rising	Rising Falling	☒	☒	☒
Hardware Trigger Delay	0 ms	[0-999]	☒	☒	☒
Internal Illuminator	Normal	Off Normal mode Power mode	☒	☒	☒
External Illuminator	Off	Off On	☒	☒	☒
Output mode	Disabled	Disabled Toggle Busy-ready Error Part detect Part pass Part fail Vision Tool dependent Logical Tool dependent	Disabled Toggle Busy-ready Error Part detect Part pass Part fail	Disabled Toggle Busy-ready Error Vision Tool dependent Logical Tool dependent	Disabled Toggle Busy-ready Error Vision Tool dependent Logical Tool dependent
Output Duration	10 ms	[0-1000] ms	☒	☒	☒
Output Delay	Disabled	Disabled [1-1000] ms	☒	☒	☒

Parameters	Default values	Values	OBJ	AOR	ID / PRO
Logical Outputs	Positive	Positive Negative	☒	☒	☒
Image Saving	None	None All If part detect If part pass If part fail Control result Logical Tool result	None All If part detect If part pass If part fail	None Control result Logical Tool result	None Control result Logical Tool result
Separator	'&'	An ASCII character	-	☒	☒
Result Format	All	Values depend on Inspection.	-	☒	☒

(*) Configurable parameter only for PRO model.

5 DATAVS2 EASY GRAPHIC USER INTERFACE

After launching the program, the following screen is displayed:

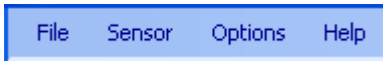
Each DataVS2 GUI component is explained separately with detail in the following chapters.

5.1 DATAVS2 GUI COMPONENTS

This chapter describes the following DataVS2 GUI components:

- Menu Bar
- Tool Bar
- Setup Panel
- Control Panel
- Image buffer dimension
- Image Panel
- Output Setup Panel
- Statistics Panel
- Inspection Explorer
- Help Panel
- Status Bar

5.1.1 Menu Bar

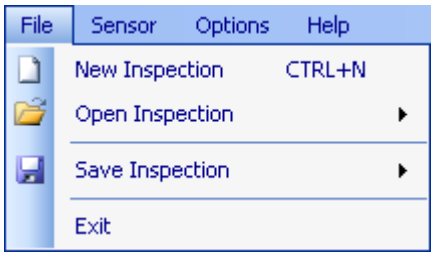


The Menu bar groups the main DataVS2 GUI functionalities in 4 different menu items:

- **File**: groups Task selection and management functions
- **Sensor**: groups the main sensor options;
- **Options**: groups the DataVS2 GUI software options;
- **Help**: shortcut to the operation manual and to DataVS2 GUI general information.

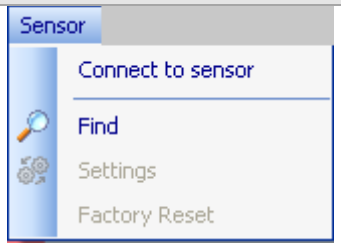
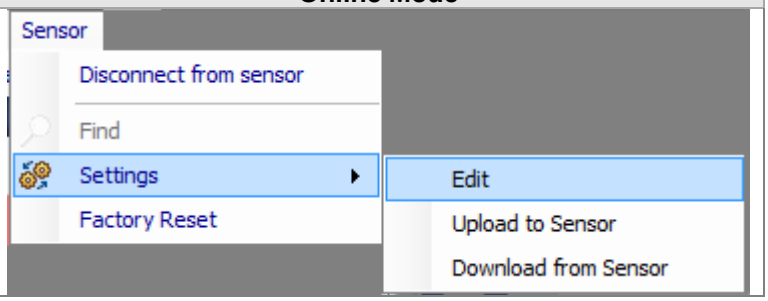
Following a description of each menu item.

File



Parameters	Description
New Inspection	Creates a New Inspection. If an Inspection is already opened, the DataVS2 GUI software asks for saving before closing it, then it resets the panels to the initial condition.
Open Inspection	Loads an Inspection into the DataVS2 GUI graphic interface. The following options are possible: • Open Inspection from PC: the Inspection to load is stored on the PC. • Open Inspection from sensor: the Inspection to load is stored in a memory slot of the sensor. If an Inspection is already opened, the DataVS2 GUI graphic interface asks for saving before closing it, then it resets panels to the initial condition.
Save Inspection	Saves the current Inspection shown by the DataVS2 GUI graphic interface. The following options are possible: • Save Inspection on PC: the Inspection is saved in a folder chosen by the user on the PC. • Save Inspection on PC (vSDK Compatibility Mode): the Inspection is saved in a folder chosen by the user on the PC in a format compatible with vSDK API. • Save Inspection on sensor: the Inspection is saved in a memory slot on the sensor. • Download All Inspections to PC: all the inspections present in the DataVS2 memory are saved in a folder chosen by the user on the PC.
Exit	Closes the DataVS2 GUI software. If an Inspection is already opened, the DataVS2 GUI graphic interface asks for saving before closing.

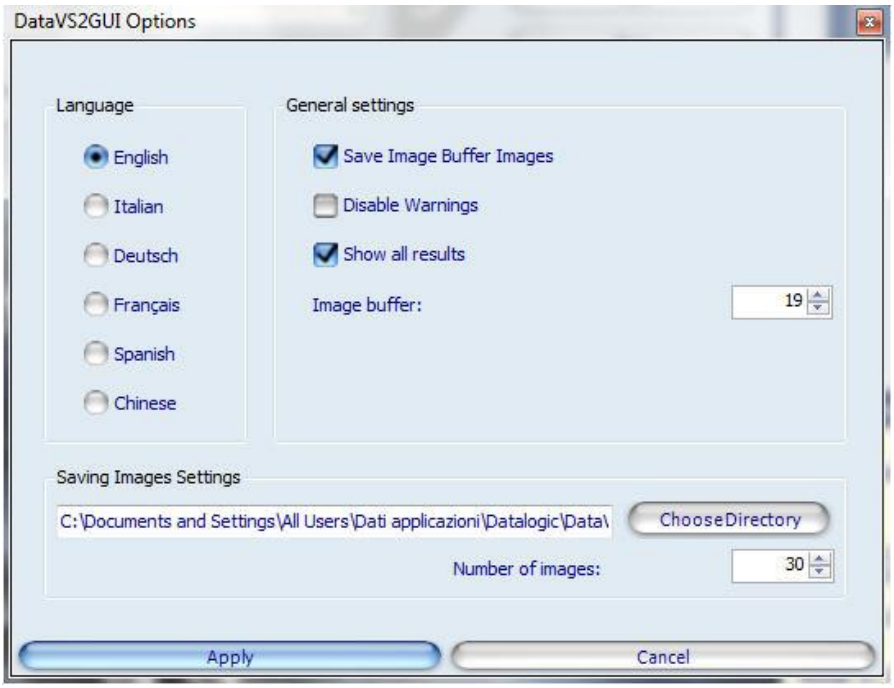
Sensor

Offline Mode	Online Mode
	

Parameters	Description	Operation Mode
Connect to sensor	Establish a connection with sensor through Ethernet	Offline
Find	Searches for and find sensor in a network	Offline
Disconnect from sensor	Disconnect the DataVS2 GUI from the current connected sensor	Online
Settings → Edit	Open the Sensor Settings window, allowing the user to change general sensor settings and manage firmware update.	Online
Settings → Upload to Sensor	It allows to load on the connected sensor the general settings previously saved in a file. After the file selection the following window is displayed: Note: it is possible to use this functionality with sensors with fw version equal or higher than ST2.4.0.04. Note: it is necessary that the sensor model from which the general sensor settings are saved is the same sensor model to which are uploaded.	Online
Settings → Download from Sensor	It allows to save the general settings of the connected sensor to a file on your PC.	Online
Factory Reset	Reset the general default settings on the DataVS2 sensor (Chapter **3.4**). ***Note***: this function is available in ST2.3.0.01 firmware, and latter versions, of DataVS2.	Online

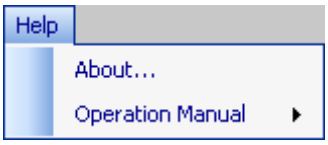
Options

The Options menu displays the following popup window that allows you to configure the DataVS2 GUI options:



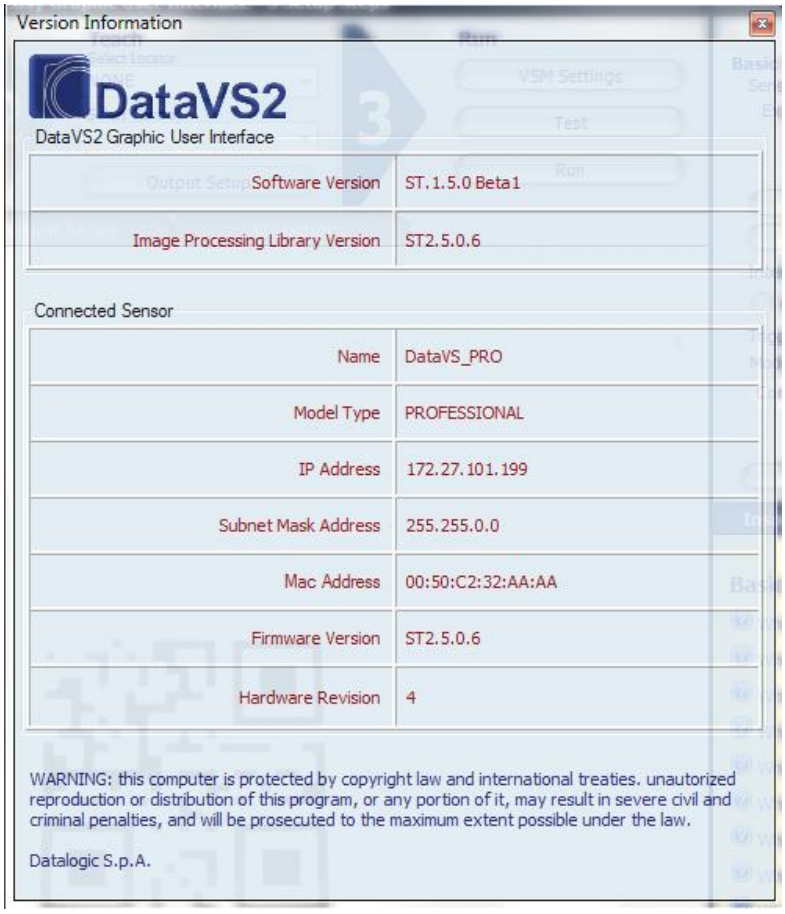
Parameters	Description
Language	The currently selected language in the DataVS2 GUI software. Five are the languages managed: • English • Italian • German • French • Spanish • Chinese After selecting the desired language, press the “Apply” button to confirm. Pressing the “Cancel” button will restore the previous condition.
Save Images in Buffer	When saving the Inspection on the PC, if this option is enabled, all the images currently present in the Image Buffer will be saved together with the Inspection on the PC. ***Note***: to see images saved on the PC you will need to open the Inspection in the DataVS2 GUI graphic interface. Images are saved in a compressed format.
Disable Warnings	If enabled, warning messages for Pattern Match, 360° Pattern Match and 360°Contour Match Tool configuration will not be displayed in STEP2.
Show all results	If enabled all the graphic results of the 360° Pattern Match Tool are shown in STEP 3. We suggest to disable this flag to speed up the DataVS2 GUI when in test mode and to assure that all images are downloaded from sensor.
Image buffer dimension	The capacity of the Image buffer: the number of images that can be stored at the same time in the image buffer.
Image saving options: directory	Use this function to set the directory where the current Inspection images are saved according to the saving condition defined in Setting panel Run→Expert. The image saving function is active both in the Test mode as well as in the Run mode.
Image saving options: maximum number of saved images	Sets the maximum number of images saved in the directory; once full, the following images overwrite the previous ones starting from the oldest. If the number is set to 0, there is not a maximum value of images in the folder, so the images will never be overwritten.
Apply changes	Applies changes to the DataVS2 GUI.
Cancel	Restores previous options. All changed options will be ignored by the DataVS2 GUI.

Help



Parameters	Description
Operation Manual	Displays the DataVS2 and VSM user manuals; each manual is displayed by DataVS2 GUI in a separate window.
About	Shows information on the DataVS2 GUI graphic interface and connected sensor version (see image below).

The following window opens when the “About” menu is pressed:



Parameters (from top to bottom)	Description
Software version	The Graphic User Interface software version.
Image Processing Library version	The DataVS2 GUI integrated simulator version. ***Note***: after connecting the sensor, the DataVS2 GUI graphic interface compares the simulator version with the currently connected sensor version; if different, a popup message will suggest to update the sensor.
Name	The name of the currently connected sensor. ***Note***: this field is empty in Offline mode.
Model type	The currently connected sensor Model Type. ***Note***: this field is empty in Offline mode.
IP address	The currently connected sensor IP Address. ***Note***: this field is empty in Offline mode.
Subnet Mask	The currently connected sensor Subnet Mask Address. ***Note***: this field is empty in Offline mode.
Mac Address	The currently connected sensor MAC Address.

	Note: this field is empty in Offline mode.
Firmware version	The currently connected sensor firmware version. ***Note***: when Firmware version is different from the Simulator version you should update the firmware on the sensor ***Note***: this field is empty in Offline mode.
Hardware Revision	The currently connected sensor Hardware revision. ***Note***: this field is empty in Offline mode.

5.1.2 Tool Bar

Gives quick access to typical functions. It is divided in 8 sections:

Section 1: name of currently loaded Inspection

In this section, a textbox allows you to configure the Name of the current Inspection.

NOTE

When a new Inspection is created, a default name is assigned by the DataVS2 GUI, "SvsInspection".

Section 2: functions related to the File menu

Function (left to right)	Description
New Inspection	Creates a New Inspection. If an Inspection is already opened, the DataVS2 GUI graphic interface asks for saving before closing it, then it resets the panels to the initial condition.
Open Inspection	Loads an Inspection into the DataVS2 GUI graphic interface. The following options are possible: • Open Inspection from PC: the Inspection to load is stored on the PC. • Open Inspection from sensor: the Inspection to load is stored in a memory slot of the sensor. If an Inspection is already opened, the DataVS2 GUI graphic interface asks for saving before closing it, then it resets panels to the initial condition.
Save Inspection	Saves the current Inspection shown by the DataVS2 GUI graphic interface. The following options are possible: • Save Inspection on PC: the Inspection is saved in a folder chosen by the user on the PC. • Save Inspection on PC (vSDK Compatibility Mode): the Inspection is saved in a folder chosen by the user on the PC in a format compatible with vSDK API. • Save Inspection on sensor: the Inspection is saved in a memory slot on the sensor. • Download All Inspections to PC: all the inspections present in the DataVS2 memory are saved in a folder chosen by the user on the PC.

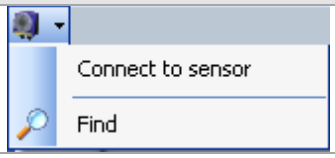
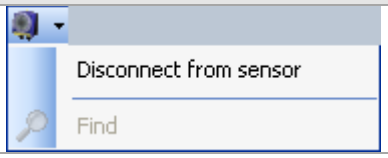
Section 3: functions related to the Image Panel

Function (left to right)	Description
Zoom Out	Changes the current Zoom Factor of the image displayed in the IMAGE PANEL, by decreasing it of 1.5 points. ***Note***: the default and minimum Zoom Factor is 1.0.
Zoom In	Changes the current Zoom Factor of the image displayed in the IMAGE PANEL, by increasing it of 1.5 points. ***Note***: the default and minimum Zoom Factor is 1.0. The maximum Zoom Factor is 5.0

Section 4: functions related to the Image Buffer

Function (left to right)	Description
Previous image	Shows the previous image in the IMAGE PANEL
Next Image	Shows the next image in the IMAGE PANEL
Add Image	Adds one or more images to the IMAGE BUFFER. The image to add is loaded from the PC.
Delete Image	Removes the current image from the IMAGE BUFFER.

Section 5: find and connect Sensor

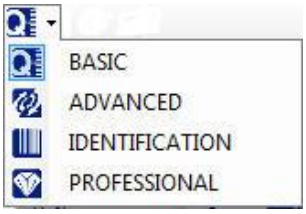
Offline Mode	Online Mode
	

Functions	Description	Operation Mode
Connect to sensor	Establish a connection with the sensor through Ethernet.	Offline
Find	Searches for and finds sensors in a network.	Offline
Disconnect from sensor	Disconnects the DataVS2 GUI graphic interface from the currently connected sensor.	Online

Section 6: DataVS2 GUI screenshot and image management

Function (left to right)	Description
Screen Shot	Saves the screen shot of the DataVS2 GUI graphic interface.
Save Image	Saves on PC the image currently loaded in the IMAGE PANEL.

Section 7: Model type selection



Function	Description	OBJ	AOR	ID	PRO
BASIC	Creates a New Inspection with Model field set to BASIC (OBJ). ***Note***: If an Inspection is already opened the DataVS2 GUI graphic interface asks for saving before closing it and starts the new task.	☒	☒ (Offline only)	☒ (Offline only)	☒ (Offline only)
ADVANCED	Creates a New Inspection with Model field set to ADVANCED (AOR). ***Note***: If an Inspection is already opened the DataVS2 GUI graphic interface asks for saving before closing it and starts the new task.	☒ (Offline only)	☒	☒ (Offline only)	☒ (Offline only)
IDENTIFICATION	Creates a New Inspection with Model field set to IDENTIFICATION (ID). ***Note***: If an Inspection is already opened the DataVS2 GUI graphic interface asks for saving before closing it and starts the new task.	☒ (Offline only)	☒ (Offline only)	☒	☒ (Offline only)
PROFESSIONAL	Creates a New Inspection with Model field set to PROFESSIONAL (PRO). ***Note***: If an Inspection is already opened the DataVS2 GUI graphic interface asks for saving before closing it and starts the new task.	☒ (Offline only)	☒ (Offline only)	☒ (Offline only)	☒

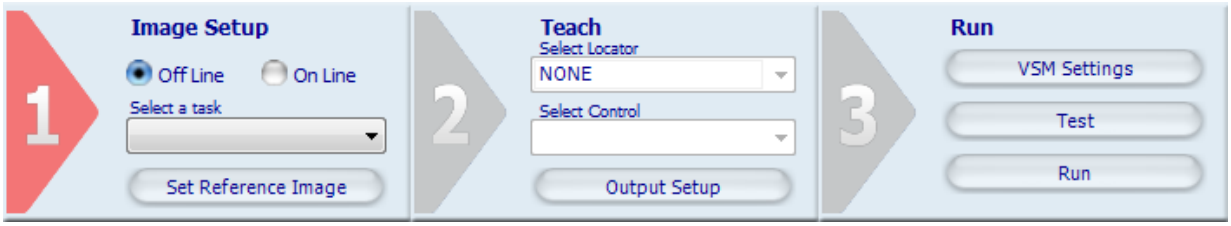
Section 8: Help

Function	Description
Manual	Opens the DataVS2 online guide

5.1.3 Setup Panel

The Setup panel lets you complete the system configuration procedure in 3 STEPS. Each STEP is identified by a number.

- **STEP1: Image Setup**. Allows you to choose the operation mode, create or load an Inspection and setup image acquisition properties.
- **STEP2: Teach**. Allows you to insert Locators, Controls and configure Output behaviour.
- **STEP3: Run**. Allows you to test the Inspection and finally run it on the sensor.



Three are the colours used by the DataVS2 GUI software to highlight the STEPS:

- **Red**: the STEP is enabled and selected.
- **Blue**: the STEP is enabled but not selected
- **Gray**: the STEP is disabled.

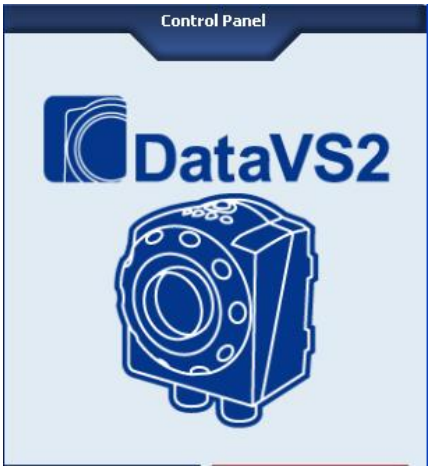
Only enabled STEPS can be selected. Only one STEP can be selected each time. Just press the triangle button to select a STEP.

5.1.4 Control Panel

The control panel offers several different functions depending on the current STEP configured and the operation you are carrying out. For instance, it allows to change the basic parameters of the sensor as well as the visualize and change the current parameters of a Locator or a Control Tool.

At the DataVS2 GUI software startup, the CONTROL PANEL is empty and displays the DataVS2 Logo.

Each section of the Operation Manual contains a description of parameters displayed in the CONTROL PANEL.



5.1.5 Image Buffer

It lists the thumbnail pictures of images which have been previously acquired from the sensor (Online mode) or loaded from the personal computer (Offline mode).

As soon as images are loaded in the DataVS2 GUI software, their thumbnails pictures are shown in the IMAGE BUFFER. The image currently displayed in the IMAGE PANEL is highlighted with a red frame inside the IMAGE BUFFER. You may switch to a different image by simply clicking on the corresponding thumbnail or by using the arrow buttons in the tool bar.

The maximum number of images currently stored in the IMAGE BUFFER can be changed from 20 to 100 in the Options Menu of the DataVS2 GUI. The default value is 20. Changing the IMAGE BUFFER capacity will generate an IMAGE BUFFER reset.

NOTE

Be careful; changing the IMAGE BUFFER capacity will affect the memory space used by the DataVS2 GUI software on the PC. All the images shown by the IMAGE BUFFER are stored in the DataVS2 GUI installation folder.

5.1.6 Work Area Tabs

The DataVS2 GUI work area is organized in three tabs, each enables the PANEL currently displayed in the central area of the DataVS2 GUI:

- **IMAGE PANEL**: displays images related to the current Inspection loaded from the sensor or from the PC.
- **OUTPUT SETUP PANEL:** displays the panel used for Logical Tools configuration. It can be selected only in STEP2.

NOTE

This panel is enabled only for AOR Inspection Models.

- **STATISTICS**: displays the panel used for displaying the Inspection execution statistics and timings values. It can be selected only in STEP3. It is enabled only in the Online mode.

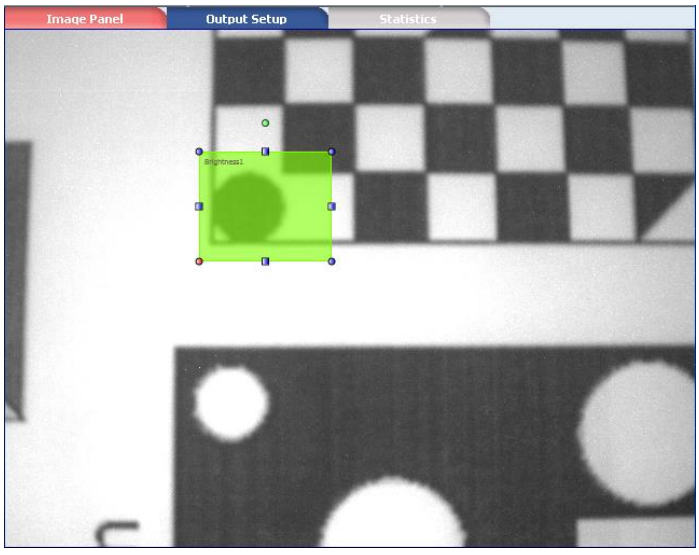
The DataVS2 GUI software uses three colours to highlight the PANELS:

- **Red**: the PANEL is enabled and selected.
- **Blue**: the PANEL is enabled but not selected
- **Gray**: the PANEL is disabled.

The DataVS2 GUI software automatically manages the PANEL visualization according to the STEP currently selected. If you want to change the selected panel, just left-click with the mouse the corresponding tab button.

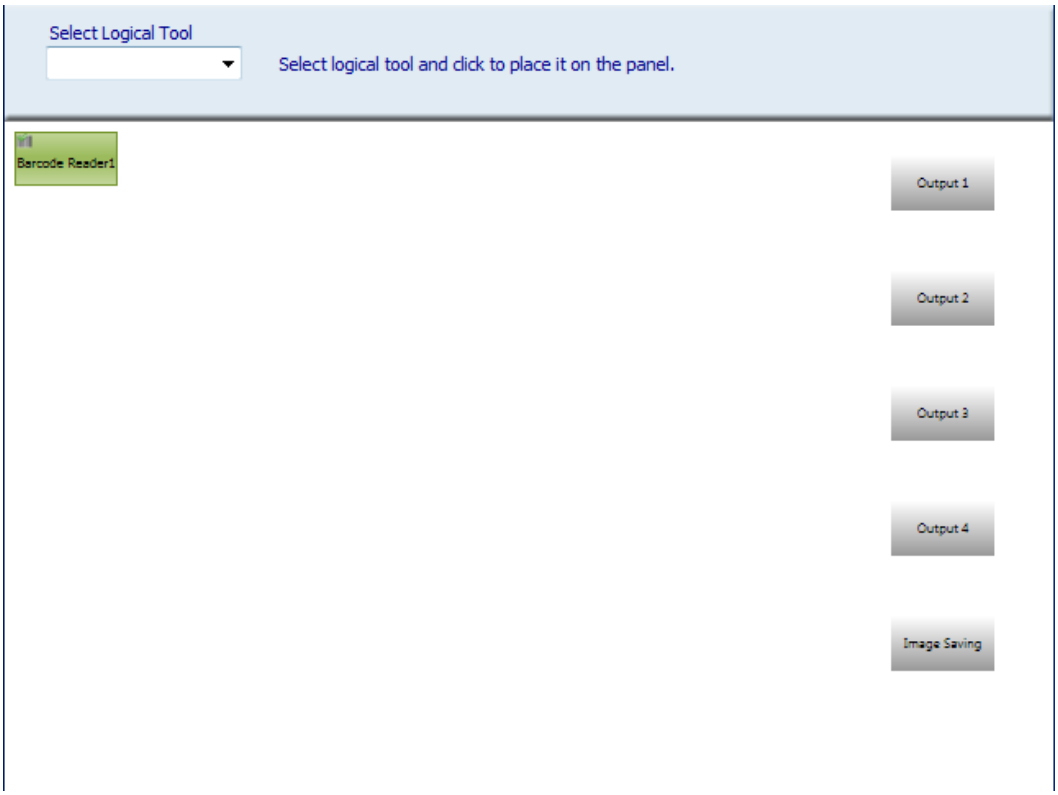
Image panel

The IMAGE PANEL displays the images acquired from the sensor when in Online mode or the images loaded from the PC when in Offline mode.
The IMAGE PANEL is enabled in STEP1, STEP2 and STEP3.
When you are in STEP2 the displayed image is the Reference Image of the current Inspection.
When in STEP 2 and STEP 3, the IMAGE PANEL shows also the ROIs of the Locator and the Controls inserted in the current Inspection.



Output Setup panel

The OUTPUT SETUP panel displays the controls for Logical Tools and output configuration when working with AOR Models.
It is automatically selected by the DataVS2 GUI software when you press the OUTPUT PANEL button in STEP2.
When no Control nor Locator are present in the current Inspection, the OUTPUT PANEL is disabled.



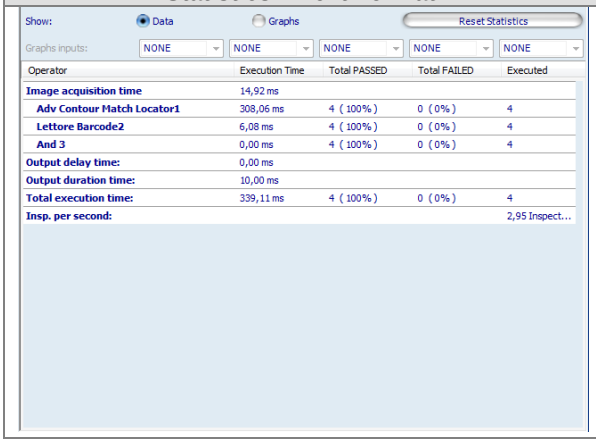
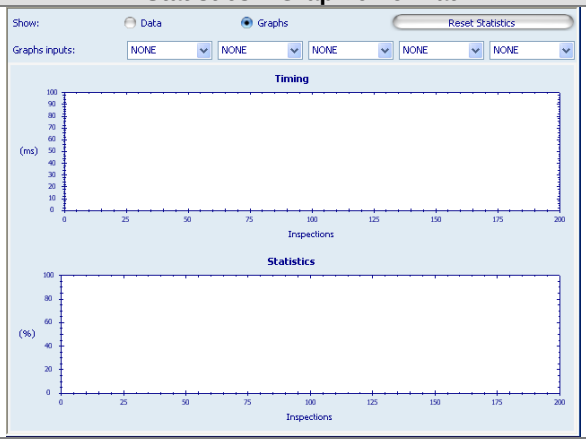
Statistics panel

The STATISTICS PANEL displays the execution time values and the statistics of the currently running Inspection when in STEP3.
The STATISTICS PANEL is automatically displayed by the DataVS2 GUI software when you are in STEP3, the RUN button is pressed and “Visualize Statistics and Timing” button in CONTROL PANEL is selected. The Inspection is running on the connected sensor and timings and statistics are loaded on the STATISTICS PANEL after each execution cycle.
When the STATISTICS PANEL is shown all other panels are disabled.

The DataVS2 GUI offers two display modes of the statistics and results: Text and Graphic.
The following table lists and describes all the parameters displayed and the functions present in the Statistics panel.

NOTE

Beginning from the ST2.3.0.01 firmware version of DataVS2, the statistics are calculated inside the DataVS2. In the previous firmware versions, the statistics were calculated on the DataVS2 GUI.

Statistics – Text Format	Statistics – Graphic Format
	

The following table lists the type of data displayed.

Parameters	Description
Text Format	In this mode all the data relative to the current Inspection running on the sensor are displayed in a tabular format: each line of the two tables displayed (one dedicated to execution time and the other dedicated to execution statistics) contains one single datum, identified by a name and corresponding value. ***Note***: The data currently displayed can be saved in a text file by selecting the field and clicking on the right button of the mouse.
Graphic Format	In this mode all the data relative to the current Inspection running on the sensor are displayed in two graphics: Execution time: the graphic shows the progression of the execution time of the selected elements with respect to the number of execution cycles completed, beginning from when DataVS2 has begun the current Inspection. Statistics: the graphic shows the percentage of successes (positive results) of selected elements with respect to the number of execution cycles completed, beginning from when DataVS2 has begun the current Inspection. To select the elements to display in the graphics described above, use the selection menus available on the top of the Statistics Panel. Up to five elements can be monitored per Inspection being carried out. ***Note***: the data currently displayed can be saved in an image file by selecting the field and clicking on the right button of the mouse. This menu allows to print and copy the displayed graphic.
Reset statistics	Resets the statistics calculated by the sensor since the beginning of the inspection (or previous reset) as soon as the button is pressed.
Image acquisition	The time required by the sensor to acquire a new image during Inspection execution. The time is represented in milliseconds.
Control execution	For each Locator, Control and Logical Tool inserted in the currently running Inspection the panel displays the logical execution result (PASS or FAIL) and the execution time in milliseconds.
Output duration time	The Output Duration value set by the user for the currently running Inspection.
Total execution time	The execution time of the currently running Inspection. It is calculated as follows: Image acquisition time + Control execution time + Output duration time.
Worst total execution time	The worst (highest) execution time performed by the sensor from the beginning of the execution (once pressed Start button in the CONTROL PANEL).
Output Delay	The output delay value set by the user for the currently running Inspection.
Inspections per second	The number of Inspections executed in one second by the sensor. ***Note***: this value is calculated by the DataVS2 GUI software according to the timing values calculated by the sensor. It refers to the Inspection results processed by the DataVS2 GUI. If the Inspection execution is too fast the DataVS2 GUI may not process all the results received by the sensor.

Parameters	Description
Total Number Inspections executed	The total number of Inspections executed by the sensor from the beginning of the execution. ***Note***: this value is calculated by the DataVS2 GUI software according to the timing values calculated by the sensor. It refers to the Inspection results processed by the DataVS2 GUI. If the Inspection execution is too fast the DataVS2 GUI may not process all the results received by the sensor.
Total Number of Inspections PASSED	The total number of Inspections executed whose result was SUCCESS from the beginning of the execution. ***Note***: this value is calculated by the DataVS2 GUI software according to the timing values calculated by the sensor. It refers to the Inspection results processed by the DataVS2 GUI. If the Inspection execution is too fast the DataVS2 GUI may not process all the results received by the sensor.
Total Number of Inspections FAILED	The total number of Inspection executed whose result was FAILURE from the beginning of the execution. ***Note***: this value is calculated by the DataVS2 GUI software according to the timing values calculated by the sensor. It refers to the Inspection results processed by the DataVS2 GUI. If the Inspection execution is too fast the DataVS2 GUI may not process all the results received by the sensor.

5.1.7 Inspection Explorer

The INSPECTION EXPLORER displays all the parameters related to the current Inspection (see par. 4.5); each parameter is updated to the value set by the user during Inspection configuration.
To view the INSPECTION EXPLORER left-click with the mouse on the related tab on the left side of the main window.

The parameters are displayed in a hierarchic view, where each of them is represented by a branch of the tree.

All the branches are grouped in 9 nodes:

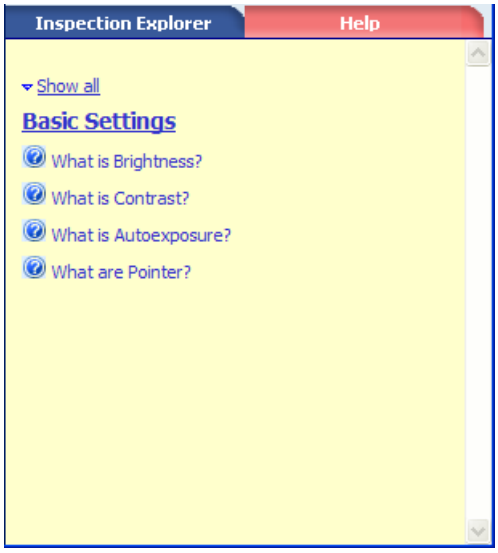
- **Inspection Name**: root node. Represents the name of the current Inspection.
- **Inspection version**: the version of the current Inspection. The Inspection version field is strictly related to the DataVS2 GUI software version used for its configuration. Inspections created with older versions of the DataVS2 GUI may have different version values.
- **Model Type**: Model Type value of the current inspection. Each time you create a New Inspection the DataVS2 GUI software asks you to select one of the available Model Types (see Chapter 13). The Model Type value corresponds to one of the sensor available model types and is used by the DataVS2 GUI software to decide if the current Inspection can be saved on the current connected sensor.
- **Sensor Settings**: groups Exposure Time, Gain and Image Resolution settings.
- **Trigger**: groups Trigger Mode, Trigger Delay and Trigger Duration values.
- **Illuminators**: groups Internal and External Illuminator modes
- **Outputs**: groups the Mode value, the Output Duration, Output Delay and Logic values for each of the three sensor outputs
- **Controls**: groups the configuration parameters and logical results for each of the Locator and Control inserted in the current Inspection. Tool nodes are red or green, depending on the current logic result.
- **Logical Tools** : contains the Type and the Name of the Tool for each Logical Tool inserted in the current Inspection. Logical Tool nodes are red or green, on the current logic result.

To see the parameters grouped inside a node of the INSPECTION EXPLORER, just left-click with the mouse on the node in order to expand it.

5.1.8 Help

The HELP PANEL displays a dynamic help which guides you through all the STEPs. It provides tips and detailed information on the following topics:

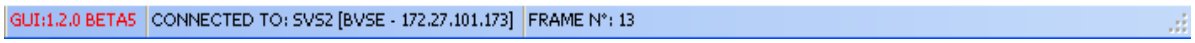
- Software overview
- Inspection setup
- Basic sensor settings
- Trigger parameters
- Inspection parameters
- Locator Controls and Logical tools parameters



5.1.9 Status Bar

It provides an instant overview of DataVS2 GUI status.
It shows:

- the DataVS2 GUI software version
- the connected sensor's Name and IP Address
- the number of images downloaded from the sensor (in LIVE, TEST and in RUN mode with Graphics enabled)
- the frame rate (in LIVE, TEST and in RUN mode with Graphics enabled)



5.2 IMAGE PROCESSING TOOLS

Image Processing Tools play a key role in the inspection process: they are used to perform controls and measurement on acquired images. Configuring the Tools is an operating step of the inspection.

Most of these Tools are positioned graphically on the IMAGE PANEL using the mouse. When you insert a Tool, the CONTROL PANEL displays the whole set of configuration parameters with their default values expressed in pixels. Positioning and sizing may be performed by changing these numeric parameters.
The execution of one of these Tools provides two types of results:

1. **Numeric value**, represents the score calculated during image processing. The numeric format used to represent the Calculated value can vary depending on the kind of Image Processing Tool:
   - Percentage: represents the Calculated Value for Photometric Tools and Matching Tools (Brightness, Contrast, Contour Match and Pattern Match).
   - Integer value: represents the calculated value (in pixels) for Measuring Tools (Position, Width and Edge Count).

2. **Status:** represents the logical result calculated during image processing. Two are the possible values for the status of an Image Processing Tool:
   - SUCCESS: Tool with gave a positive result
   - FAIL: Tool with a negative result or control/measurement could not be performed.

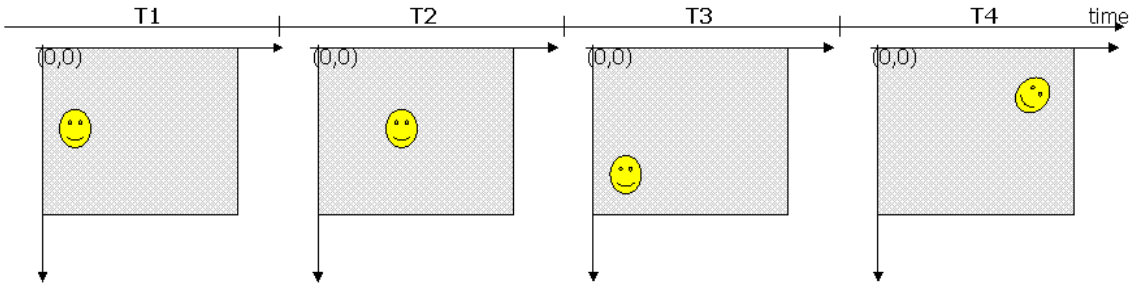
The area assigned to the controls may cover the whole image or be limited to the ROI (Region Of Interest), i.e. just a portion of the acquired image, with consequent benefits in terms of speed and evaluation accuracy.
During inspection, the area outside the ROI is ignored, although it is displayed in the IMAGE PANEL.

All the Image Processing Tools use an absolute reference system, with the origin located in the top left corner of the work area.

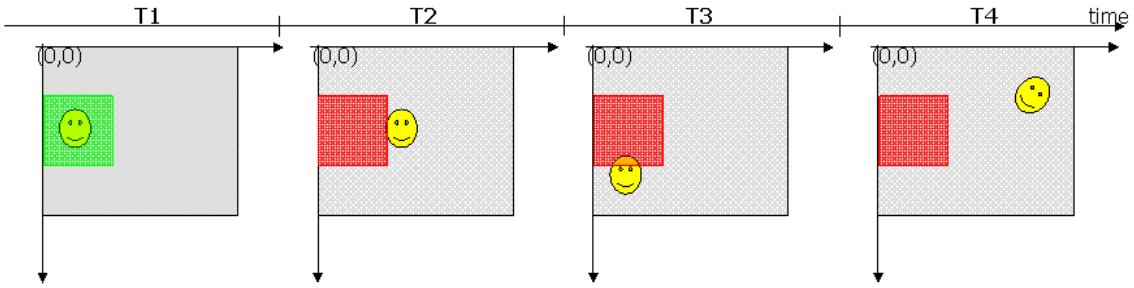
5.2.1 Locators and Controls

The DataVS2 GUI software organizes Image Processing Tools in two different categories: **Locators** and **Controls**.

The main difference between a Control and a Locator is the result meaning of the image elaboration process. A Locator is used to find the part to inspect inside the image; in a typical application the sensor is installed on a conveyor belt carrying the pieces to inspect. Let's think about the situation in which the sensor checks only one piece at a time. Due to transportation, the pieces to inspect are not always in the same position in different images (see Figure below: T1, T2, T3, T4 are consecutive acquisition times).

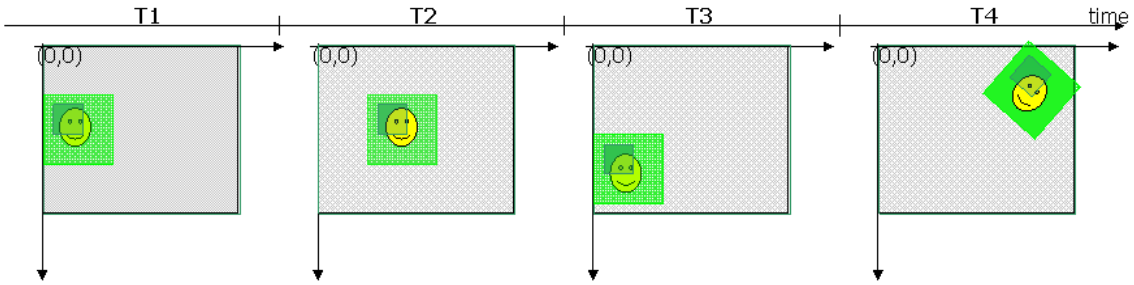


The position of the piece can affect the inspection result and can cause Inspection failure when not necessary (or vice versa), as the sensor doesn't check the right part of the piece. The following figure shows a Brightness Tool execution result: green when it is ok, red when it fails. The Brightness ROI position is fixed and referred to the absolute reference system, while the piece to inspect moves inside it from one image to another.

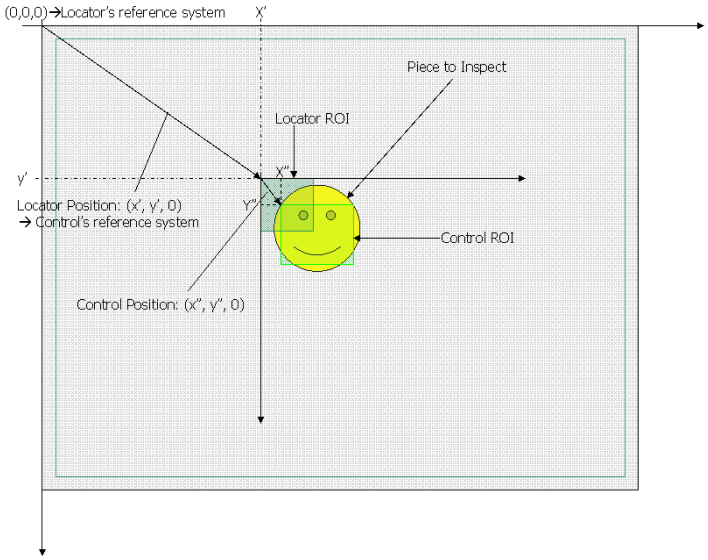


If we desire a reliable behaviour we must add a *part finder,* i.e. a locator. With the locator, the image processing cycle is organized in two phases: first the locator searches for the part and returns a position (x,y,alpha), then the Control ROIs are relocated, depending on the current position found by the locator, and Controls are executed. The result of the inspection is successful if both the locator and the controls return a result that satisfies Input conditions.

The following picture shows the locator effect (Pattern Match): the locator ROI is the little rectangle on the smiling face. During execution time, first the Locator searches for the piece inside the image, then (if the piece was found by the locator) all the Controls are relocated and referred to Locator result and then Controls are executed.



When one Locator is present, the Control ROIs position is referred to Locator Tool result (that represents the position of the piece found in the image), while Locator ROI position is fixed; the Locator result represents a relative reference system, which is the reference for Control ROIs positions (see Figure below). From the image processing algorithm point of view, a Tool is a Locator if its result can be interpreted as a position (i.e. three characteristic coordinates: [x, y, alpha]). Locator Tools can also be used as Controls, while not all Controls are Locators: only Controls returning a position can be used as Locators.



5.2.2 DataVS2 Tools

The following table describes the Tools for image processing available in the different models of DataVS2.

Tool	Description	Locator	Control	OBJ	AOR	ID	PRO
Brightness	the average intensity of pixels inside the ROI.	-	☒	☒	☒	-	☒
Contrast	Calculates the ratio between the brightest and darkest pixels inside the ROI.	-	☒	☒	☒	-	☒
Contour Match	Searches for a specific shape inside the ROI; the shape template is loaded in the Teach phase (STEP 2)	-	☒	☒	☒	-	☒
Edge Count	Counts for the number of edges inside the ROI along a searching direction passing through the ROI center.	-	☒	☒	☒	-	☒
Width	Measures the distance between two edges in pixels.	-	☒	☒	☒	-	☒
Pattern Match	Searches for a pattern inside the ROI; the pattern template is acquired in the Teach phase (STEP 2). The pattern to be found must have the same orientation of the template.	☒	☒	☒	☒	☒ (only Locator)	☒
Position	Locates the position of an edge in pixels. The edge is found searching along a direction passing through the ROI Center.	☒	☒	☒	☒	☒ (only Locator)	☒

Tool	Description	Locator	Control	OBJ	AOR	ID	PRO
360° Pattern Match	Searches for an object inside the ROI; the object template is acquired in the Teach phase (STEP 2). The object to be found may have a different orientation and angle with respect to the template.	☒	-	-	☒	-	☒
360° Contour Match	Searches for an object inside the ROI; the object template is acquired in the Teach phase (STEP 2). The object to be found may have a different orientation degree with respect to the template.	☒	-	-	-	-	☒
360° Contour Counter	Searches and counts the objects inside a ROI. The objects to be found may have a different orientation and angle with respect to the template.	-	☒	-	-	-	☒
360° Defect Finder	Detects the presence of defects comparing the object found in the ROI with the template.	-	☒	-	-	-	☒
Datamatrix	Searches and decodes one or more Datamatrix symbols in a ROI.	☒	☒	-	-	☒ (only Control)	☒
Barcode	Searches and decodes one or more Barcode symbols in a ROI.	☒	☒	-	-	☒ (only Control)	☒
OCV	Verifies the presence and print quality of a character string in a ROI.	-	☒	-	-	☒	☒
QRCode	Searches and decodes one or more QRCode symbols in a ROI.	☒	☒	-	-	☒ (only Control)	☒

Some of the most common tasks that can be solved with DataVS2 Tools are listed below:

- **Object identification** is used to distinguish different kinds of objects, e.g., to control the flow of material or to decide which inspections to perform.
- **Presence verification** is used to detect the presence of some parts in known or unknown positions, e.g., bottles inside a box, pills inside a blister, overprinting on labels.
- **Position and/or orientation detection** is used, for example, to control a robot that assembles a product by mounting the product components in the correct position.
- **Completeness checking** to ensure that the product has been assembled correctly, i.e., that the right components are in the right place.
- **Shape and dimensional inspection** is used to check the geometric parameters of a product to ensure that they lie within the required tolerances.

5.3 LOCATORS

A Locator is a special Tool which searches inside the Region Of Interest for a certain characteristic e.g. a gray value edge or a pattern and determines its position. Once the position has been calculated, it is used as the origin for the Control ROIs included in the same Inspection.

NOTE

Only ONE Locator can be present in the same Inspection.

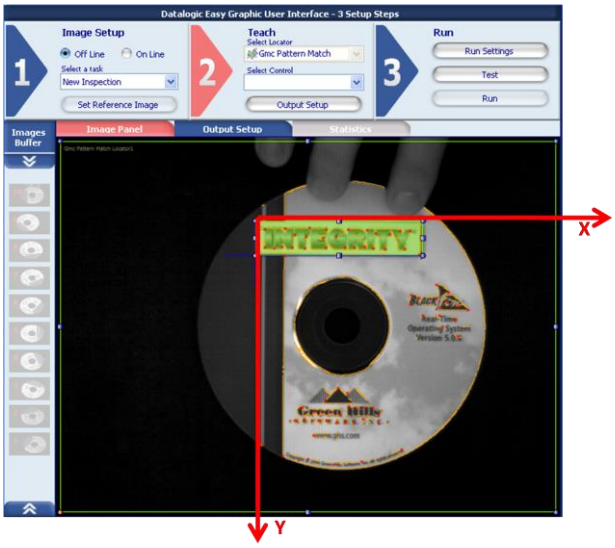
NOTE

If the Locator has not found any position in the current image or in the Reference Image, all the other Controls included in the same Inspection will NOT be executed. In this case, only the ROI of the Locator highlighted in red will be shown by the DataVS2 GUI in the IMAGE PANEL. All other Control ROIs will not be shown.

5.3.1 Usage of Locators

Without any Locator present in the Inspection, all the Controls will use the upper left corner of the image as the absolute, stationary origin for their ROI positions.

This means that, even if the object in the current image undergoes displacements respect to the object in the Reference Image, the ROI position will NOT change. As a result, the object may not be properly tested. We recommend to use a Locator with all Inspections, unless the object is stationary and has no rotational, horizontal or vertical displacements. If a Locator is present, all the Controls will use the absolute position returned by the Locator as the origin for their ROIs positions. Thus the Control's ROI position changes if the Locator has found the feature it was searching for, e.g. the CD logo as shown in the following picture.



5.3.2 Selection and Positioning

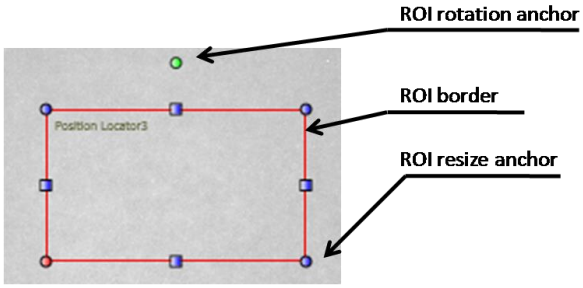
Depending on the DataVS2 model, (see the table in par. 5.2.2), several types of Locators may be selected from the pull-down menu under the “Select Locator” label in STEP 2:

- Position Locator
- Pattern Match Locator
- Geometric Pattern Match Locator
- Contour Match 360°
- Datamatrix Locator
- Barcode Locator
- QRCode Locator

NOTE

If you move your mouse cursor in the IMAGE PANEL after you have selected a Locator, it will change appearance; it will show the icon of the Locator as shown in the pull-down menu.

After you have selected the Locator, move your mouse inside the IMAGE PANEL. Now click on any position on the image. This will create a ROI frame as shown in the following image (where a Position Locator ROI is shown).



If you want to change the ROI size, left-click and drag one of the blue anchors you find on ROI borders. If you want to move the ROI, place the cursor over one of the 4 ROI borders; it will change its icon into a 4 crossed-arrowed one. Press and hold the left mouse button, and drag the rectangle to the desired position if the image.
If you want to rotate the ROI, left-click and drag the green anchor on the top of the ROI: the ROI will be rotated around its barycentre.

After adding the Locator to the current Inspection the DataVS2 GUI CONTROL PANEL will show the relative parameters.

NOTE

Not all Tools allow carrying out all the modifications on the relative ROIs (ROI Moving, ROI Resizing, ROI Rotation).

5.3.3 Position Locator

It is also known as Edge locator. An Edge is a defined border between a light area and a dark area in the image.
The Position Locator identifies the transition between:

- A light and a dark area (**White to Black**)
- A dark and a light area (**Black to White**)
- in the image analyzed.

For instance, imagine a dark object on a light background: object contour is identified as the edge, i.e. the set of dark pixels lying next to light pixels.
However, a sharp tone variation for the light-to-dark transition must be present to be interpreted as an edge, measured by the “threshold”. (Edge detection ►)

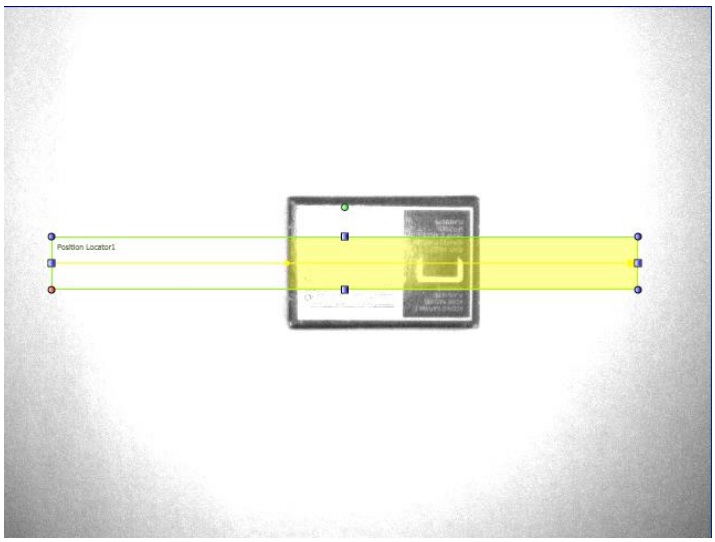
Position Locator selection and positioning

You can select the Position Locator from the Locators in the pull-down menu in STEP2. The Position Locator is identified by the following icon:

When you are in STEP 2, the Position Locator ROI allows the following variations:

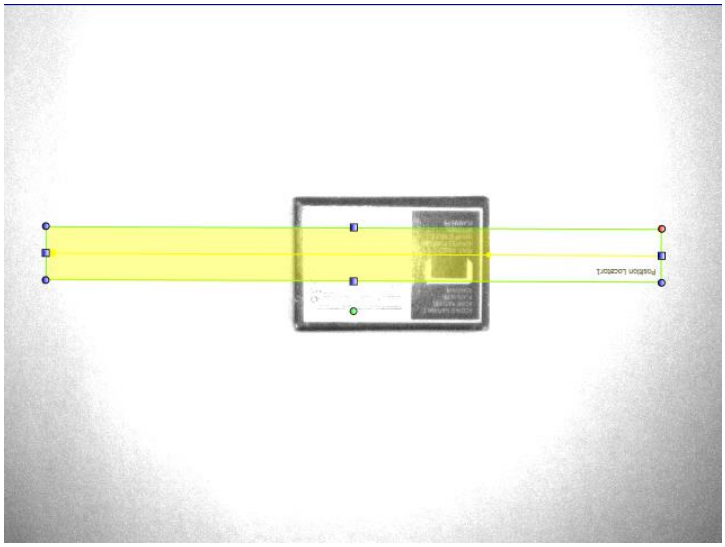
- ROI Moving: the position of the ROI can be changed inside the Reference Image.
- ROI resizing: the width and the height of the ROI can be changed inside the Reference Image.
- ROI Rotation: the orientation of the ROI can be changed inside the Reference Image (only in the DataVS2 GUI 1.2 version).

The edge detected is graphically represented inside the ROI with a yellow dot located in the matching point. The searching direction and versus is represented by a yellow arrow passing through the ROI barycentre and going from Left to Right.
If the edge position matches the “Position Limits” range, the ROI borders will be shown in green, representing a SUCCESS condition.
If no edges are found or if the position found is outside “Position Limits”, the ROI borders will be shown in red, representing a FAILURE condition.
After, the Variation Type of the edge (White to Black or Black to White) to be identified has been specified, the Locator processes the ROI in the direction shown by the yellow arrow and stops as soon as it comes across an edge that meets the specifications.



NOTE

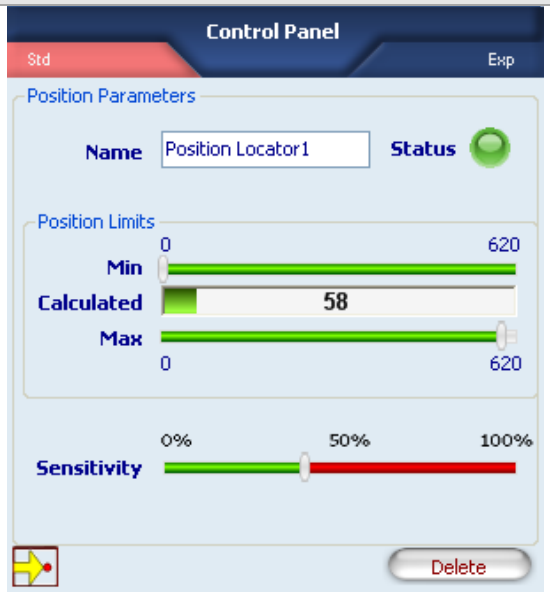
The searching direction represented by the yellow line is always referred to the ROI's top-left corner. When the ROI orientation is 180 degree you will see the yellow line going from Right to Left.



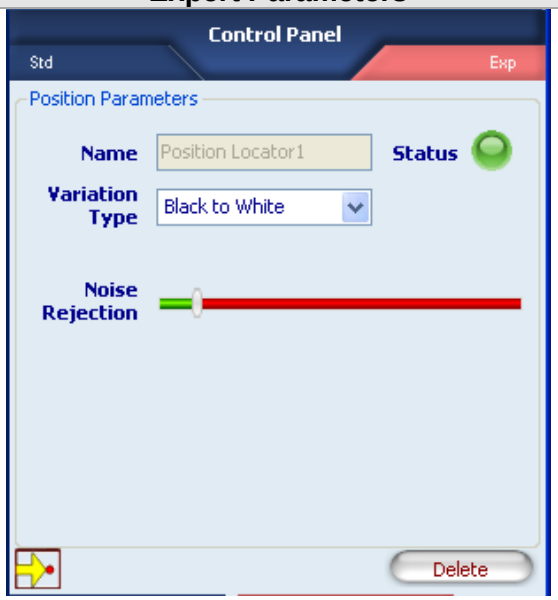
Position Locator parameters

The following image and the table display the Position Locator parameters.

Standard Parameters



Expert Parameters



Parameters	Description	Type	Default
Name	The Name assigned to the Locator Tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	Position Locator

Parameters	Description	Type	Default
Status	Tool that indicates a result: • Green: SUCCESS. The Position Locator has detected an Edge within the range set in “Position limits”. • Red: FAILURE: Position Locator hasn’t found any edge or the found Edge doesn’t match “Position limits”	Standard, Output	
Calculated	Current position of the edge found by the position Locator inside the ROI along the searching direction.	Standard, Output	
Minimum	Minimum value of detectable position inside the ROI, along the searching direction. This threshold must be compatible with the value found (shown near the "Calculated" label): this means Min must be less than or equal to "Calculated".	Standard, Input	0
Max	Maximum value of detectable position inside the ROI, along the searching direction. This threshold must be compatible with the value found (shown near the "Calculated" label): this means Max must be more than or equal to "Calculated". Note: this means Min must be less than or equal to "Calculated. 0 (Min) is always considered the left border of the ROI, the value shown as Max is always considered the right border of the ROI and represent the ROI width in pixels. Looking at the picture shown above ROI width is 620 pixels.	Standard, Input	ROI Width
Sensitivity	The degree of accuracy provided when searching for the Edge. The lower the Sensitivity, the greater the accuracy.	Standard, Input	40%
Variation Type	Determines the type of edges to be searched inside the image. Four are the available options: • From white to black: only white to black transitions are detected; • From black to white: only black to white transitions are detected; • Auto: both white to black and black to white transitions are detected; the returned one is the transition with the largest difference between the gray values. • Both: both white to black and black to white transitions are detected; the returned one is the first transition found.	Expert, Input	All
Noise Rejection	Reduces the effect that noisy pixels will have on the edge detection. The higher the value, the stronger the brightness variation necessary to recognize an edge.	Expert, Input	5
Delete	Deletes the Locator from the current Inspection		

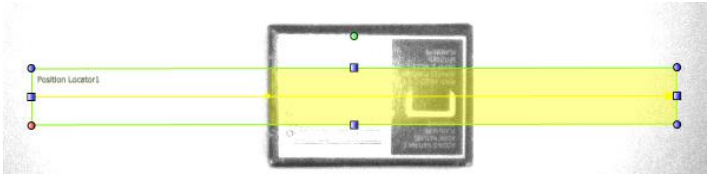
Example

- Add a position locator to current inspection.
- Select "Position" from the pull-down menu:

- Position the control on the reference image in the IMAGE PANEL: the pointer changes shape as soon as it enters in the work area. Left-click once on any position to create a default rectangular ROI.
- At the same time, the control characteristics and the result of its application on the reference image appear in the control panel.
- Set ROI size and position as required, as explained previously.

- The result indicated by STATUS is also highlighted by the ROI border changing colour.
- As soon as values change, the result is calculated without delay.

- A yellow horizontal bar appears in the ROI with a small dot marking the position of the identified edge. In this example, the edge was set as light-to-dark transition. If the edge is not recognized correctly, move the slider to adjust Sensitivity. Or you may switch to EXPERT mode and change the advanced parameter "Noise Rejection".



- Another application example is shown in the figures below, Fig. A and Fig. B. In particular, the Fig. A shows how the Position Locator is forced to detect a horizontal translation of a box among the scene. As shown, the locator application shifts the control tool to the right and the inspection process is correctly achieved.

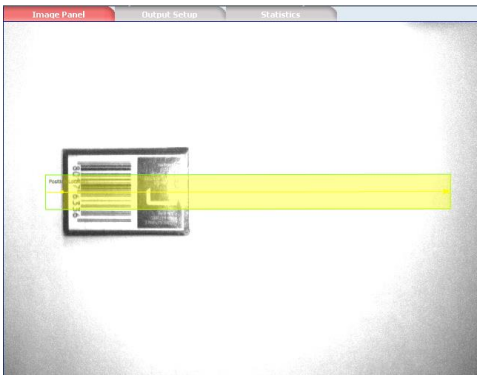


Fig. A CD Position Locator: Reference image

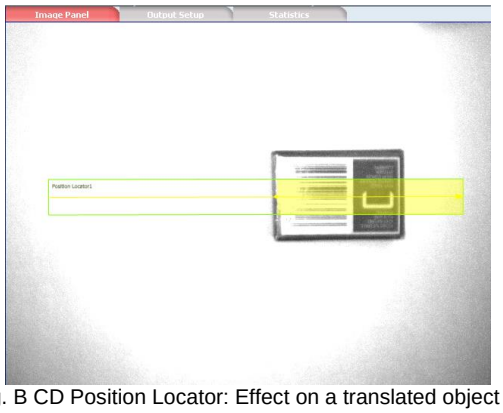


Fig. B CD Position Locator: Effect on a translated object to the right

5.3.4 Pattern Match Locator

The pattern match operation verifies that the target object and current object are similar. (Pattern match ►).

The Pattern Match Locator searches for the target inside a well-specified area of the image and determines its position. In this way it is possible to detect the horizontal translations and the vertical displacements of the object in the inspection area.

The Pattern Match Locator finds horizontal and vertical displacements in the SEARCH AREA. It can be used to compensate the rotational displacement up to a maximum of 5-10 degrees.

Pattern Match Locator: selection and positioning

You can select the Pattern Match Locator from the Locators in the pull-down menu in STEP2. The Pattern Match Locator is identified by the following icon:

Pattern Match Locator is characterized by two different ROIs, having different scopes:

- SEARCH AREA: the outer rectangle, representing the portion of the image where the Locator will search for the target object.
- TEMPLATE ROI: the inner rectangle, representing the target pattern to be searched in the region delimited by the search area. The TEMPLATE ROI cannot be positioned outside the SEARCH AREA.

When you are in STEP 2, the Pattern Match Locator ROI allows the following variations:

- Move the SEARCH AREA and the TEMPLATE ROI: the position of the ROI can be changed in the Reference Image.
- SEARCH AREA and TEMPLATE ROI resizing: the width and the height of the ROI can be changed in the Reference Image
- SEARCH AREA rotating: the orientation of the Search Area can be changed inside the Reference Image. The rotation of the Search Area involves also the ROI rotation of the Template. Rotating the Search Area, you can notice that the Template ROI will rotate accordingly.

During the setup process, you have to define the target object to be identified. The system stores the brightness value of each composing pixel included in the TEMPLATE ROI.
During the inspection, the locator will search for the target in the SEARCH AREA. When it finds the target, it will highlight it with a green rectangle.
The degree of similarity and accuracy may be set from the control panel.

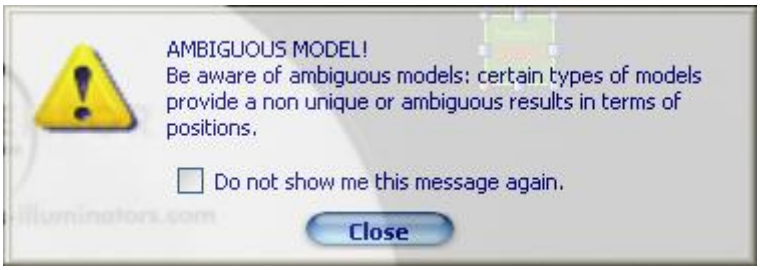
NOTE

In order to have a reliable and robust behaviour of the Pattern Match Locator, we recommend to configure the TEMPLATE ROI using a well-contrasted reference image of the target object.

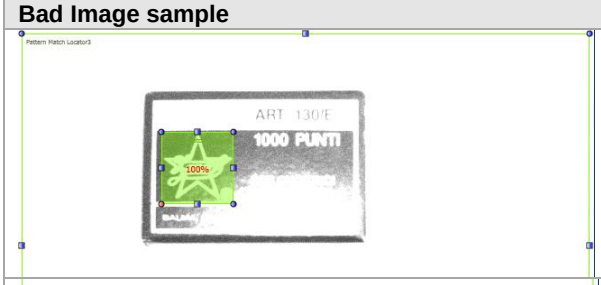
Some useful hints that can help you choose a good target object image:

- The reference image has to be in focus and must have a good contrast and brightness level. The user can adjust image brightness and contrast by changing the image acquisition Exposure Time and Gain in STEP1.
- The target object has to be completely included inside the template ROI borders; adjust TEMPLATE ROI dimensions and position in order to include only the object to be searched.
- The target object texture should be recognizable and in contrast with its background; assure that the Reference image doesn't contain ambiguous objects or presenting the same texture of the target object. Only the target object should be present in the reference image while the remaining part of the image (the background) should be uniform.

The DataVS2 GUI software helps you in define a good quality TEMPLATE ROI: a popup message will be shown whenever the chosen target object is unsuitable (see picture below).
When this message is shown, please try to adjust the TEMPLATE ROI by resizing or moving it. If nothing changes, we suggest you to choose a different reference image.

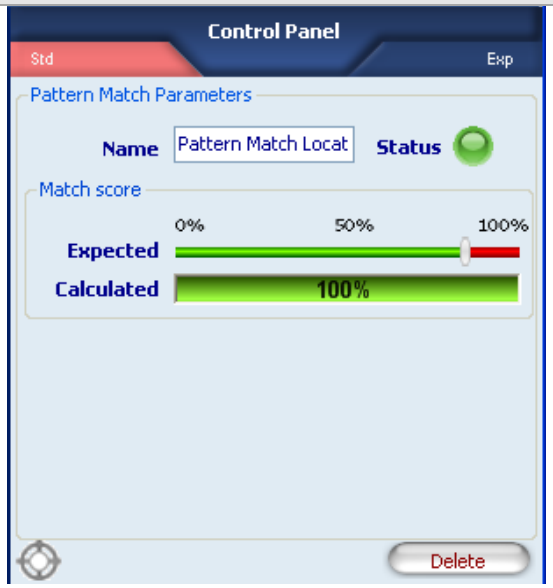
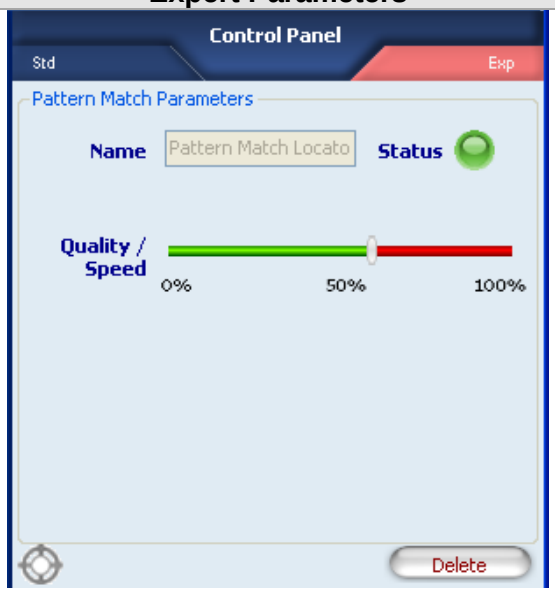


The following table shows some examples of a bad target object configuration:

Bad Image sample	Description
	The Reference Image is not in focus and doesn't a good contrast and brightness level. In this case you should change the Reference Image after adjusting focus and increasing the Exposure Time.
	The target object (sensor Logo) is not completely included inside the TEMPLATE ROI borders. The dimensions of the template ROI should be adjusted in this case.
	The target object texture cannot be distinguished enough from its background. In this case you should change the TEMPLATE ROI after modifying the gain; this should increase the contrast in the image

Pattern Match Locator parameters

The following image and the table display Pattern Match Locator parameters.

Standard Parameters	Expert Parameters
	

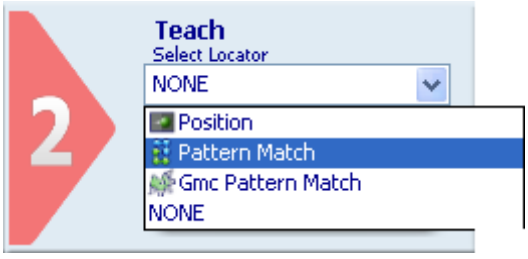
Parameters	Description	Type	Default
Name	The Name assigned to the Locator Tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by an univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	Pattern Match Locator
Status	A tool result indicator . • Green: SUCCESS. The Pattern Match Locator has found the target pattern with a score higher that the expected one. • Red: FAILURE. Pattern Match Locator hasn't found the target pattern or the score of the found pattern is lower than the expected one.	Standard, Output	
Calculated	The score representing the calculated degree of similarity of the current found object to the reference target one.	Standard, Output	
Expected	Threshold used for the result evaluation. It indicates how closely the found pattern in the SEARCH AREA must resemble the target pattern in order to be considered valid. 100% means perfect match, 0% means no match found.	Standard, Input	85%
Accuracy / Speed	Influences the precision level used by the Pattern Match Tool during the searching phase; the higher the accuracy value, the higher the precision in the calculated score of the found object. Low level of Accuracy will speed up the searching process: we recommend to use low values of Accuracy when the execution time doesn't meet your requirements.	Expert, Input	60%
Delete	Deletes the Locator from the current Inspection		

To quicken the Pattern Match Locator execution time you can act on two different parameters:

- Reducing the SEARCH AREA dimensions: if you are sure that the object to be found is always in a restricted area of the image, we recommend to reduce SEARCH AREA dimension in order to cover only the portion of the image where you want the object to be searched. This will reduce the time needed to locate the object (as the portion of the image where to search for the target pattern is smaller) and will decrease the risk of errors due to the noise inside the image.
- Reducing the Accuracy value: reducing the level of precision required to the Pattern Match Tool during the searching phase, will decrease the time needed to calculate the match score.

Example

- Add a Pattern Match locator to the current inspection.
- Select "Pattern Match" from the pull-down menu:



- Position the control on the reference image in the IMAGE PANEL: the pointer changes shape as soon as it enters in the work area. Left-click once on any position to create a default rectangular ROI.
- At the same time, the control characteristics and the result of its application on the reference image appear in the control panel.
- Set ROI size and position as required, as explained previously.
- The result indicated by STATUS is also highlighted by the ROI border changing colour.
- As soon as the values change, result is calculated without delay.
- Hereafter, the Figure A and B show a working example of the Pattern Match Locator. In this particular case, the Pattern Match Locator searches for the Logo position.

Fig. A Pattern Match Locator: Reference image

Fig. B Pattern Match Locator: Effect of a translated object

5.3.5 360° Pattern Match Locator

In pattern match operation, target object and current object are compared to determine if they are similar.

The Geometric Pattern Match Locator is able to detect the position and the orientation of a target object inside a well specified area of the image. In this way it is possible to detect the horizontal and vertical translations, either rotational displacement or of the object among the inspection area. The Geometric Pattern Match Locator can be used to compensate rotational displacement up to 360 degrees.

360° Pattern Match Locator: selection and positioning

You can select the Geometric Pattern Match Locator from the Locators pull-down menu in STEP2. The Geometric Pattern Match Locator is identified by the following icon:

Geometric Pattern Match Locator is characterized by two different ROIs, having different scopes:

- SEARCH AREA: the outer rectangle, representing the portion of the image where the Locator will search for the target object.
- TEMPLATE ROI: the inner rectangle, representing the target pattern to be searched in the region delimited by the search area. The TEMPLATE ROI cannot be positioned outside the SEARCH AREA.

When you are in STEP 2 the Pattern Match Locator ROI allows for the following variations:

- Move the SEARCH AREA and the TEMPLATE ROI: the position of the ROI can be changed inside the Reference Image.
- SEARCH AREA and TEMPLATE ROI resizing: the width and the height of the ROI can be changed inside the Reference Image

During the setup process, you need to define the target object to be identified. The system stores the features characterizing the target to recognize, corners and contours. A corner is an intersection between two edges found inside the image, a contour is a segment or a line corresponding to an edge. Corners are represented by the DataVS2 GUI software with red squares inside the image, while contours are represented as orange curves. Corners are more relevant than contours for the 360° Geometric pattern Match locator.

When configuring the 360° Geometric Pattern Match you should assure that all corners identifying the object are included in the TEMPLATE ROI. Moreover the higher the number of corners identifying the object the more precise will be the position of the object found by the 360° Geometric Pattern Match.
During the inspection, the locator will search for the target in the SEARCH AREA. When it finds an instance of the target, it will highlight it with a green rectangle.
Degree of similarity and accuracy may be set from the control panel.

NOTE

Target objects with less than 4 corners (red squares) will not be recognized.

NOTE

In order to have a reliable and robust behaviour of the Pattern Match Locator we recommend you to configure the TEMPLATE ROI using a well contrasted reference image of the target object.

Some useful hints that can help you choose a good target object image:

- the reference image has to be in focus and have a good contrast and brightness level; you can adjust image brightness and contrast by changing the image acquisition Exposure Time and Gain in STEP1. Assure that the highlighted contours (orange curves) are placed on the target object to recognize and not on its background.
- The target object has to be completely included inside the template ROI borders; adjust TEMPLATE ROI dimensions and position in order to include only the object to be searched. Assure that the highlighted corners (red squared dots) are located uniformly on the target object. Assure at least 4 corners are present on the target object.
- The target object texture should be recognizable and in contrast with its background; assure the reference image doesn't contain ambiguous objects or features having the same texture of the target object. Only the target object should be present in the reference image while the remaining part of the image (the background) should be uniform. Assure that the highlighted contours are placed on the target object to recognize and not on its background and assure that all corners are on the target object.

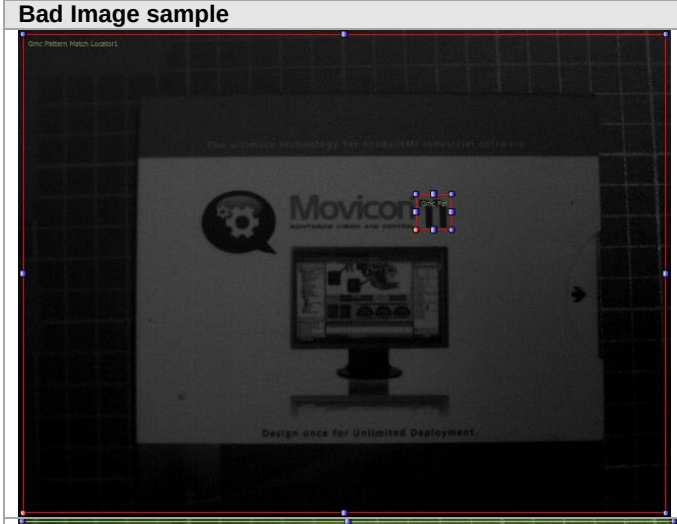
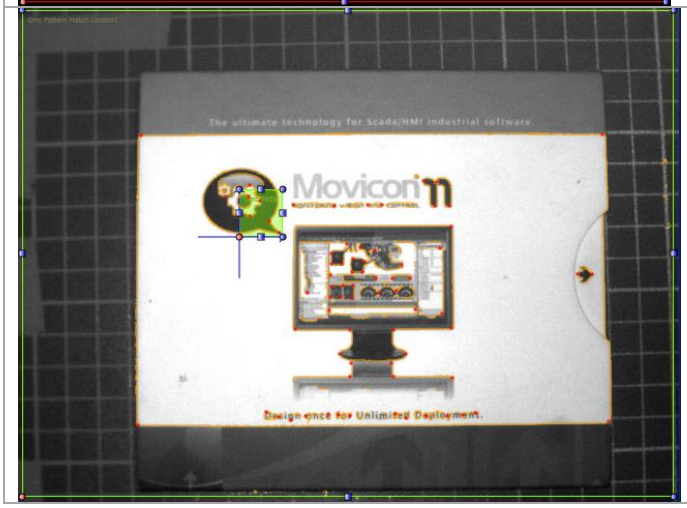
The DataVS2 GUI software helps you in defining a good quality TEMPLATE ROI: a popup message will be shown whenever the chosen target object is not good (see picture below).

When this message is shown, please try to adjust the TEMPLATE ROI by the mean of resizing or moving it. If nothing changes, we suggest you to choose a different reference image.

NOTE

A new version of 360° Pattern Match has been introduced beginning from the ST2.3.0.01 firmware version of DataVS2. In case of DataVS2 updating from a ST2.2.1.XX firmware version or lower than ST2.3.0.01 or higher we suggest to verify the setup settings of all the inspections present on the sensor and containing the 360° Pattern Match tool.

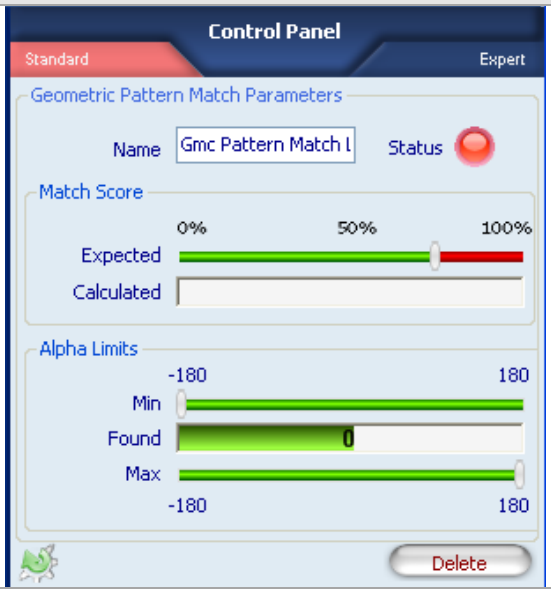
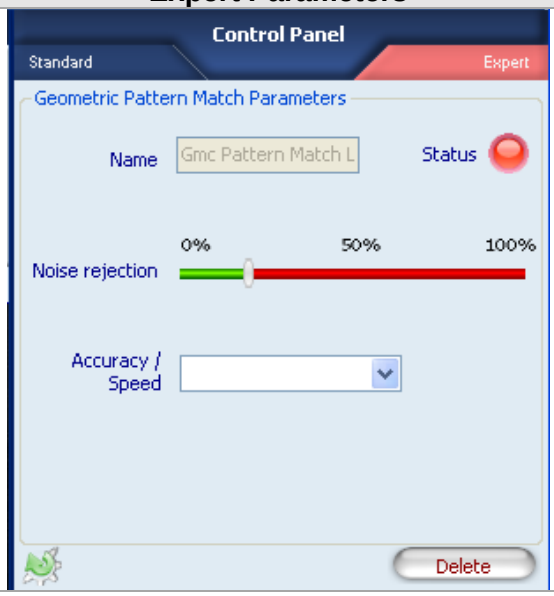
In the following table you will find some examples of a bad target object configuration:

Bad Image sample	Description
	The Reference Image is not in focus and hasn't a good contrast and brightness level. In this case you should change the Reference Image after adjusting focus and increasing the Exposure Time.
	The target object (sensor Logo) is not completely included inside the TEMPLATE ROI borders. The dimensions of the template ROI should be adjusted in this case.

	The target object texture is not recognizable enough compared with its background. In this case you should change the TEMPLATE ROI after modifying the gain; this should increase the contrast in the image

360° Pattern Match Locator parameters

The following image and the table display the 360° Pattern Match Locator parameters.

Standard Parameters	Expert Parameters
	

Parameters	Description	Type	Default
Name	The Name assigned to the Locator Tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	360° Pattern Match Locator
Status	A tool result indicator . Green: SUCCESS. The Pattern Match Locator has found the target pattern with a score higher that the expected one. Red: FAILURE. Pattern Match Locator hasn't found the target pattern or the score of the found pattern is lower than the expected one.	Standard, Output	
Calculated	The score representing the calculated degree of similarity of the current found object to the reference target one.	Standard, Output	
Expected	Threshold used for the result evaluation. It indicates how	Standard,	75%

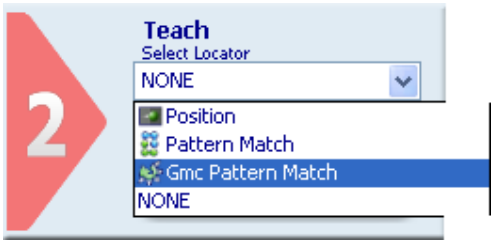
Parameters	Description	Type	Default
	closely the found pattern in the SEARCH AREA must resemble the target pattern in order to be considered valid. 100% means perfect match, 0% means no match found.	Input	
Found (Angle limit)	The orientation degree of the found target ***Note***: the orientation degree of the found object is always referred to the interval [-180°, +180°].	Standard, Output	
Min	The minimum orientation degree that the found object must have in order to return SUCCESS. If found object has an orientation degree lower than minimum expected the tool will fail.	Standard Input	-180°
Max	The maximum orientation degree that the found object must have in order to return SUCCESS. If found object has an orientation degree higher than the maximum expected the tool will FAIL.	Standard Input	+180°
Noise Rejection	Use this parameter to reduce the number of corners not belonging to the target object or not related to characteristic element of the target object. This is useful when the image is not well contrasted or present some noise (shadows and dirty).	Expert, Input	20%
Mode (Accuracy / Speed)	Represents the level of accuracy used by the 360° Pattern Match Tool in searching the target. In Accuracy mode the Tool performs a two-phase processing: the first one to locate the object, the second one to accurately define its match score. In Speed mode the Tool only performs the first phase of the processing; the match score in this case is calculated in the first phase. In Speed mode the Tool is faster. ***Note***: we recommend to set this parameter to Accuracy when you have to perform quality controls or when the Reference Image presents a non-uniform background; Speed mode has to be used for location purposes only and when the target object is well defined in comparison with its background. ***Note***: the Tool automatically sets the best value for Accuracy/Speed parameter depending on the Reference Image characteristics.	Expert, Input	
Delete	Deletes the Locator from the current Inspection		

To quicken the Pattern Match Locator execution time you can act on two different parameters:

- Reducing the SEARCH AREA dimensions: if you are sure the object to be found is always in a restricted area of the image, we recommend you to reduce SEARCH AREA dimension in order to cover only the portion of the image where you want the object to be searched. This will reduce time needed to locate the object (as the portion of the image where searching for the target pattern is smaller) and will decrease the risk of errors due to noise inside the image.
- Setting the Accuracy /Speed parameter to Speed: reduce the Pattern Match Tool precision level requested during the search phase will reduce the time necessary to calculate the score.

Example

- Add a 360° Pattern Match locator to current inspection.
- Select "360° Pattern Match" from the pull-down menu:



- Position the control on the reference image in the IMAGE PANEL: the pointer changes shape as soon as it enters the work area. Left-click once at any position to create a default rectangular ROI.
- At the same time, the control features and the result of its application to the reference image appear in the control panel.
- Set ROI size and position as required, as explained above.
- The result indicated by STATUS is also highlighted by the ROI border changing colour.
- As soon as values change, result is calculated without delay.
- Hereinafter, the Figure A and B show a working example of the Geometric Pattern Match Locator. In this particular case, the Geometric Pattern Match Locator search for the CD Logo position.

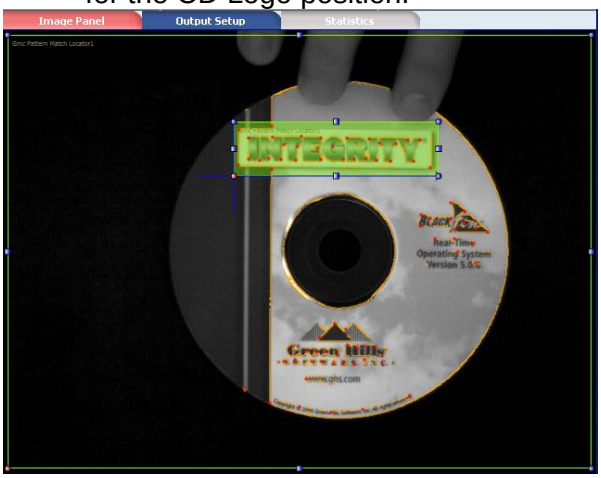


Fig. A Pattern Match Locator: Reference image

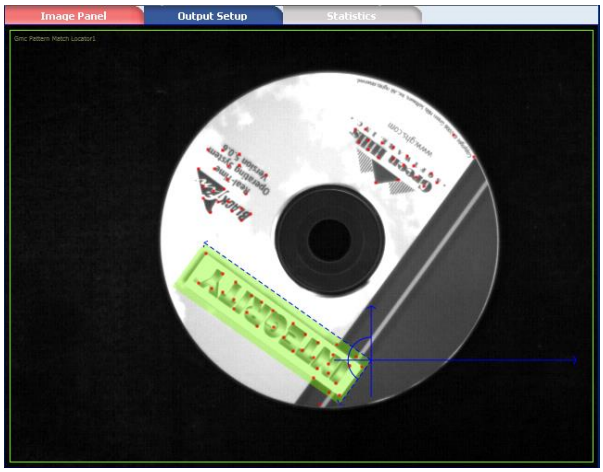


Fig. B Pattern Match Locator: Object rotation and shift

5.3.6 360° Contour Match Locator

(only available in DataVS2 PRO Model)

The Locator is able to detect the position and the orientation of a target object inside a well specified area of the image.

360° Contour Match Locator: selection and positioning

You can select the 360° Contour Match Locator from the Locators pull-down menu in STEP2. The 360° Contour Match Locator is identified by the following icon:

The 360° Contour Match Locator is characterized by two different ROIs, having different scopes:

- SEARCH AREA: the outer rectangle represents the portion of the image where the Locator will search for the target object.
- TEMPLATE ROI: the inner rectangle represents the object template to be searched in the region delimited by the SEARCH AREA. The TEMPLATE ROI cannot be positioned outside the SEARCH AREA.

When you are in STEP 2, the 360° Contour Match Locator allows for the following variations:

- Move the SEARCH AREA and the TEMPLATE ROI: the position of the ROI can be changed inside the Reference Image.
- SEARCH AREA and TEMPLATE ROI resizing: the width and the height of the ROI can be changed inside the Reference Image

During the setup process, you need to define the target object to be identified. The system memorizes the characteristics of the object to detect, represented by the object contours.
An Edge in a digital image is a geometrical area of a group of pixel characterized by the same gray level and have adjacent pixel elements with a lower or higher gray level.

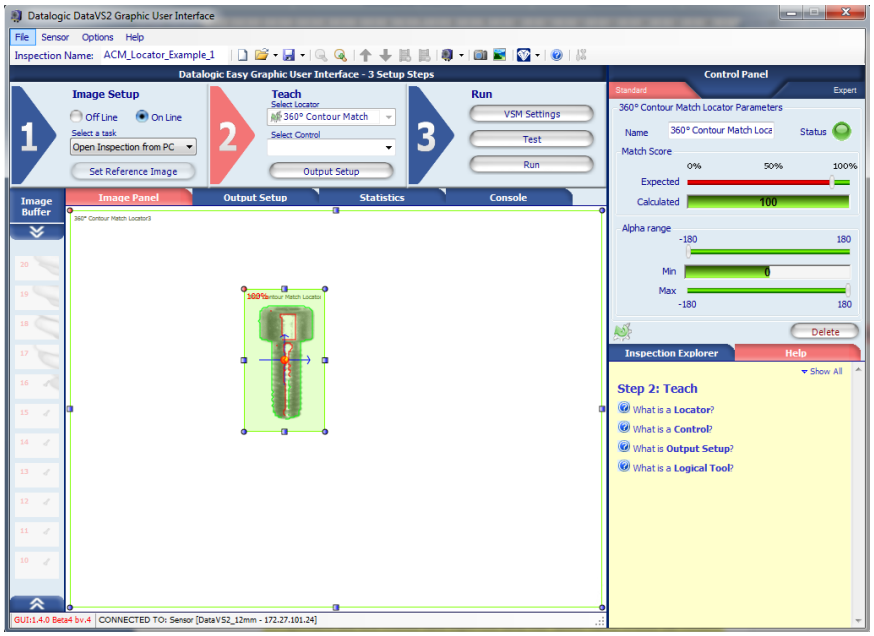
360° Contour Match Locator: contour configuration

When the 360° Contour Match Locator is configured, the object to detect must be characterized by clear contours, very evident inside the reference image.
For this reason, during the configuration of the TEMPLATE ROI, the group of contours found can be modified by selecting the good ones and deleting the less representative contours of the object from the template.

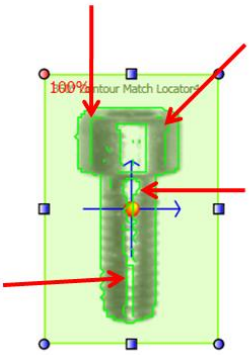
The fundamental steps of the template configuration are listed below.

1. Verify the contours acquired on the template: after positioning the TEMPLATE ROI on the reference image, the system highlights the contours found inside it. These are represented by green curves visualized inside the TEMPLATE ROI. To avoid acquiring contours not belonging to the template ensure that the TEMPLATE ROI contains only the desired target. If the reference image presents areas with variable brightness or reflections generated by the object itself, not meaningful contours can be detected. In this case remove noise.

   The following image shows the configuration phase of a 360° Contour Match locator for the detection of a metal screw.

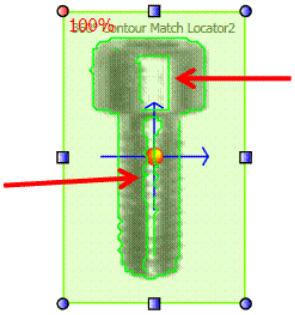


The red arrows in the following figure highlight meaningless contours, corresponding to the light reflected from the screw surface. These will be removed from the template.

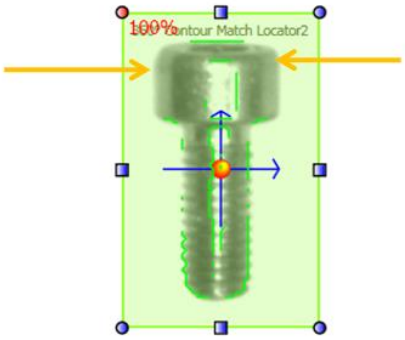


2. Noise removal: contours generated by weaker reflections can be removed using the “Noise Removal” parameter in the Expert panel of the 360° Contour Match locator parameters. In general, the higher the value of this parameter, the lower the number of contours found inside the TEMPLATE ROI.

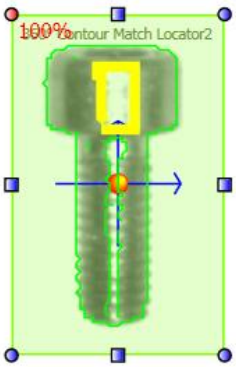
   The following figure shows the effect of this parameter on the example: spurious weaker contours have been eliminated.

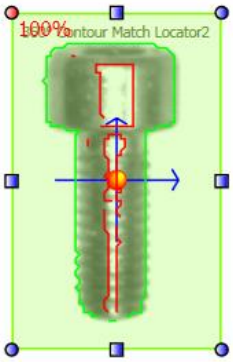


NOTE *When contours are removed, pay attention not to remove meaningful object contours. The following figure shows the effect of an excessive increase of the "Noise Removal" threshold: some meaningful object contours have been eliminated.*



3. Manual removal of meaningless contours: if at end of the noise removal, some meaningless contours are still present, the user can remove them manually. To proceed manually, the user must select each contour to remove with the left mouse button. When aimed, the contour to remove will be highlighted in yellow. When selected, the contour to remove will be represented in red.





NOTE *If some contours have been erroneously eliminated, they can be reintroduced selecting them again with the mouse.*

4. Selection of desired resemblance and search angle: once ended the contour configuration, the user can select the resemblance degree using the "Expected" parameter.

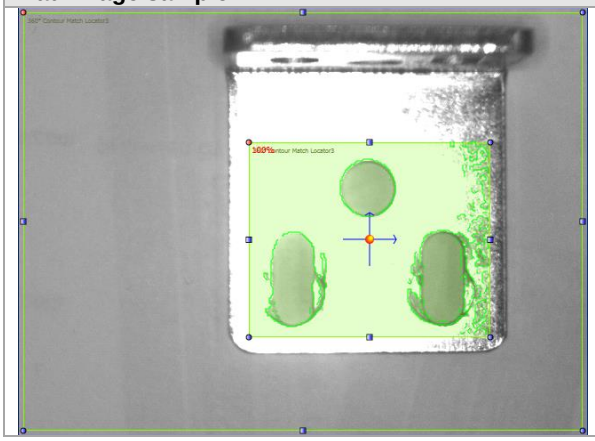
NOTE *In order to have a reliable and robust behaviour of the Contour Match Locator we recommend you to configure the TEMPLATE ROI using a well contrasted reference image of the target object.*

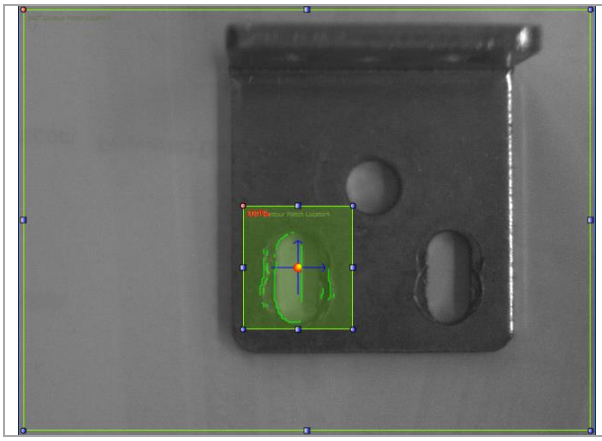
Some useful hints can help you choose a good target object image:

- The reference image has to be in focus and have a good contrast and brightness level; you can adjust image brightness and contrast by changing the image acquisition Exposure Time and Gain in STEP1. Assure that the highlighted contours (orange curves) are placed on the target object to recognize and not on its background.
- The target object has to be completely included inside the template ROI borders; adjust TEMPLATE ROI dimensions and position in order to include only the object to be searched. Assure that the highlighted corners (red squared dots) are located uniformly on the target object. Assure at least 4 corners are present on the target object.
- The target object texture should be recognizable and in contrast with its background; assure the reference image doesn't contain ambiguous objects or features having the same texture of the target object. Only the target object should be present in the reference image while the remaining part of the image (the background) should be uniform. Assure that the highlighted contours are placed on the target object to recognize and not on its background and assure that all corners are on the target object.

The DataVS2 GUI software helps you in defining a good quality TEMPLATE ROI: a popup message will be shown whenever the chosen target object is not good (see picture below).
When this message is shown, please try to adjust the TEMPLATE ROI by the mean of resizing or moving it. If nothing changes, we suggest you to choose a different reference image.

In the following table you will find some examples of a bad object contour configuration:

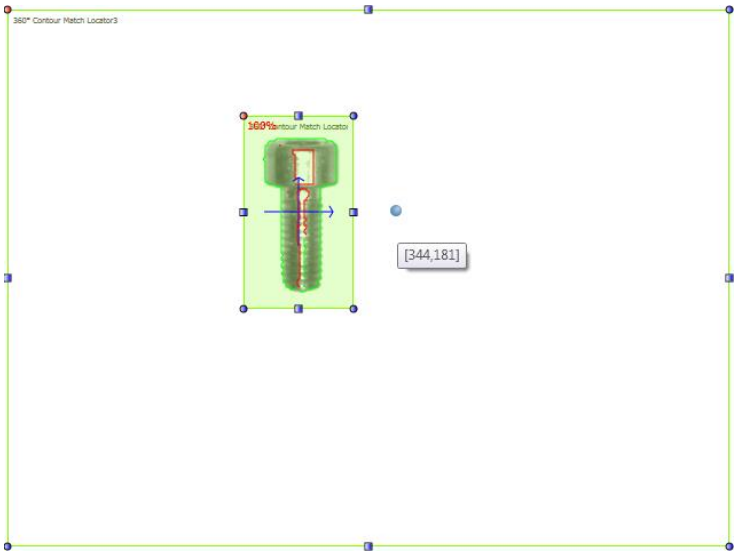
Bad Image sample	Description
	The reference image does not present homogeneous lighting; the light reflections produced by the metal surface of the object can generate instability of the object contours. In this case we suggest to change the image acquisition parameters (Exposure Time and Gain) until a more uniform image is obtained.

	The TEMPLATE ROI is not configured correctly; the highlighted object characteristics (the contours) are not sufficient to detect it, as they are not unique. When an object is characterized by repetitive geometrical elements (elliptic holes in the example) we suggest to include them all in the template.

360° Contour Match Locator: configuration of object acquisition point

The object acquisition point represents the position of the object with respect to an absolute reference system centred in the corner on the top left side of the image. It is made up of a group of Cartesian coordinates (X, Y) corresponding to the barycentre of the green rectangle that identifies the object detected in the image.

During the configuration of the 360° Contour Match locator, an acquisition point different from the default one can be defined. The user has to select the red anchor, visualized in the ROI center, with the mouse and drag it in the desired position. The new position defined for the red anchor represents the object position with respect to the green rectangle.

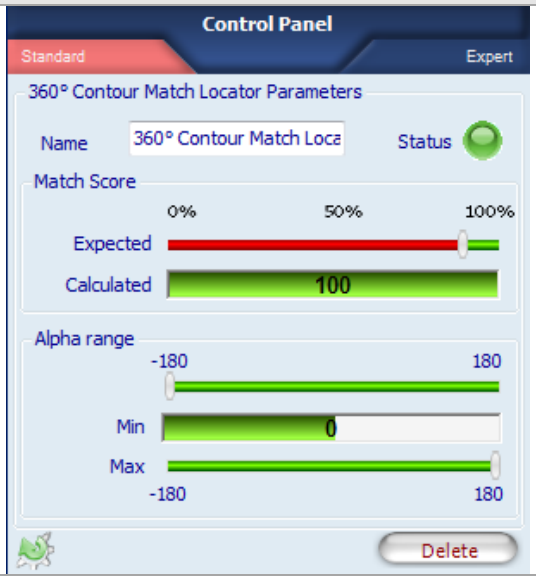
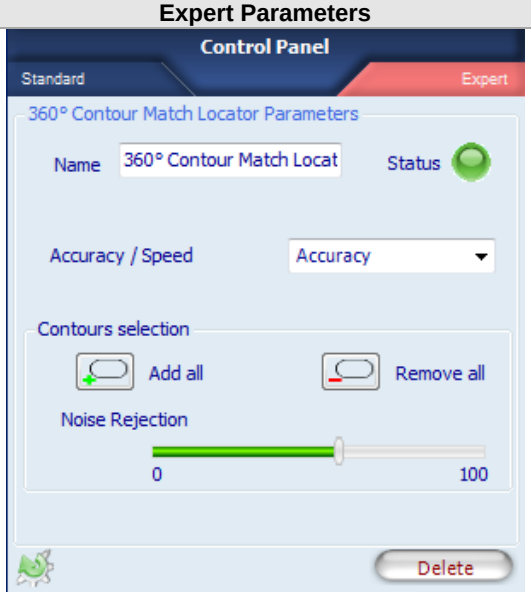


NOTE

The sensor returns as position found the one corresponding to the acquisition point.

360° Contour Match Locator parameters

The following image and the table display the 360° Pattern Match Locator parameters.

Standard Parameters	Expert Parameters
	

Parameters	Description	Type	Default
Name	The Name assigned to the Locator Tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	360° Contour Match Locator
Status	A tool result indicator . • Green: SUCCESS. The Pattern Match Locator has found the target pattern with a score higher that the expected one. • Red: FAILURE. Pattern Match Locator hasn't found the target pattern or the score of the found pattern is lower than the expected one.	Standard, Output	
Calculated	The score representing the calculated degree of similarity of the current found object to the reference target one.	Standard, Output	
Expected	Threshold used for the result evaluation. Indicates how much the object found in the SEARCH AREA must resemble the reference object to be considered valid. 100% means perfect match, 0% means no match found.	Standard, Input	75%
Search angle	Allows to set the limits on the orientation angle used to search the object inside the SEARCH AREA. • **Calculated**: the orientation degree of the found target object. ***Note***: the orientation degree of the found object is always referred to the interval [-180°, +180°]. • **Minimum**: The minimum orientation degree that the found object must have in order to return SUCCESS. If found object has an orientation degree lower than minimum expected the tool will fail. • **Maximum**: The maximum orientation degree that the found object must have in order to return SUCCESS. If found object has an orientation degree higher than the maximum expected the tool will FAIL.	Standard, Input	Minimum: -90° Maximum: 90°
Noise Rejection	Use this parameter to reduce the number of contours not belonging to the target object or not related to the elements characterizing the target object. This is useful when the	Expert, Input	20%

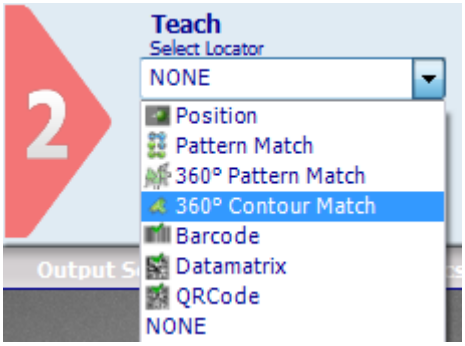
Parameters	Description	Type	Default
	image is not well contrasted or present some noise (shadows and dirty).		
Add all	Press this button to include all the contours found in the reference template. After pressing this button, all the contours in the image are visualized in green.	Expert, Input	
Delete all	Press this button to eliminate all the contours found in the reference template. After pressing this button, all the contours in the image are visualized in red.	Expert, Input	
Mode (Accuracy / Speed)	Represents the level of accuracy used by the 360° Contour Match Tool in searching the target. In Accuracy mode the Tool performs a two-phase processing: the first one to locate the object, the second one to accurately define its match score. In Speed mode the Tool only performs the first phase of the processing; the match score in this case is calculated in the first phase. In Speed mode the Tool is faster. ***Note***: we recommend to set this parameter to Accuracy when you have to perform quality controls or when the Reference Image presents a non-uniform background; Speed mode has to be used for location purposes only and when the target object is well defined in comparison with its background. ***Note***: the Tool automatically sets the best value for Accuracy/Speed parameter depending on the Reference Image characteristics.	Expert, Input	
Delete	Deletes the Locator from the current Inspection		

To quicken the Pattern Match Locator execution time you can act on two different parameters:

- Reducing the SEARCH AREA dimensions: if you are sure the object to be found is always in a restricted area of the image, we recommend you to reduce SEARCH AREA dimension in order to cover only the portion of the image where you want the object to be searched. This will reduce time needed to locate the object (as the portion of the image where searching for the target pattern is smaller) and will decrease the risk of errors due to noise inside the image.
- Setting the Accuracy /Speed parameter to Speed: reduce the Pattern Match Tool precision level requested during the search phase will reduce the time necessary to calculate the score.

Example

- Add a 360° Contour Match locator to current inspection.
- Select "360° Contour Match" from the pull-down menu:



- Position the control on the reference image in the IMAGE PANEL: the pointer changes shape as soon as it enters the work area. Left-click once at any position to create a default rectangular ROI.
- At the same time, the control features and the result of its application to the reference image appear in the control panel.

- The ROI and contours must be configured according to the requirements following the modes explained previously.
- The result indicated by STATUS is also highlighted by the ROI border changing colour.
- As soon as values change, result is calculated without delay.
- Hereinafter, the Figure A and B show a working example of the Contour Match Locator. In this particular case, the Contour Match Locator searches the position of the screw inside the image.

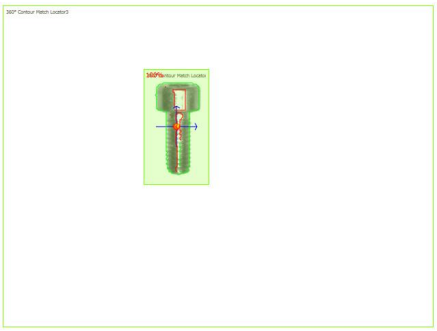


Fig. A Contour Match Locator: Reference image

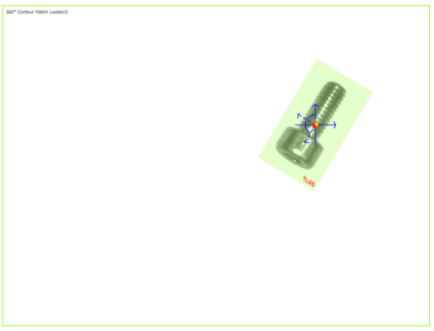


Fig. B Contour Match Locator: Object rotation and shift

5.3.7 Datamatrix Locator

(only available in DataVS2 PRO Model)

The Datamatrix Locator is able to detect the position and the orientation of one Datamatrix symbol inside a well specified area of the image.

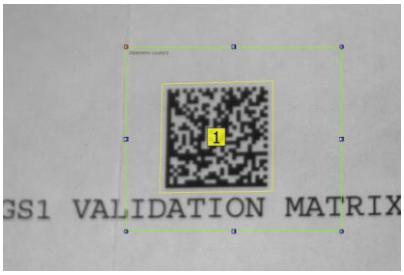
Datamatrix Locator: selection and positioning

You can select the Datamatrix Locator from the drop-down Locators menu in STEP2. The Datamatrix Locator is identified by the following icon:

When you are in STEP 2 the Datamatrix Locator allows you to make the following changes to the ROI:

- ROI Moving: the position of the ROI can be changed in the Reference Image.
- ROI Resizing: the width and the height of the ROI can be changed in the Reference Image.

The symbol found is graphically represented inside the ROI with a yellow frame that identifies the vertexes and a rectangle containing the index of the code found, visualized in the centre of the code itself.



Datamatrix Symbol Characteristics

The DataVS2 Datamatrix Locator decodes the Datamatrix symbol with the following characteristics:

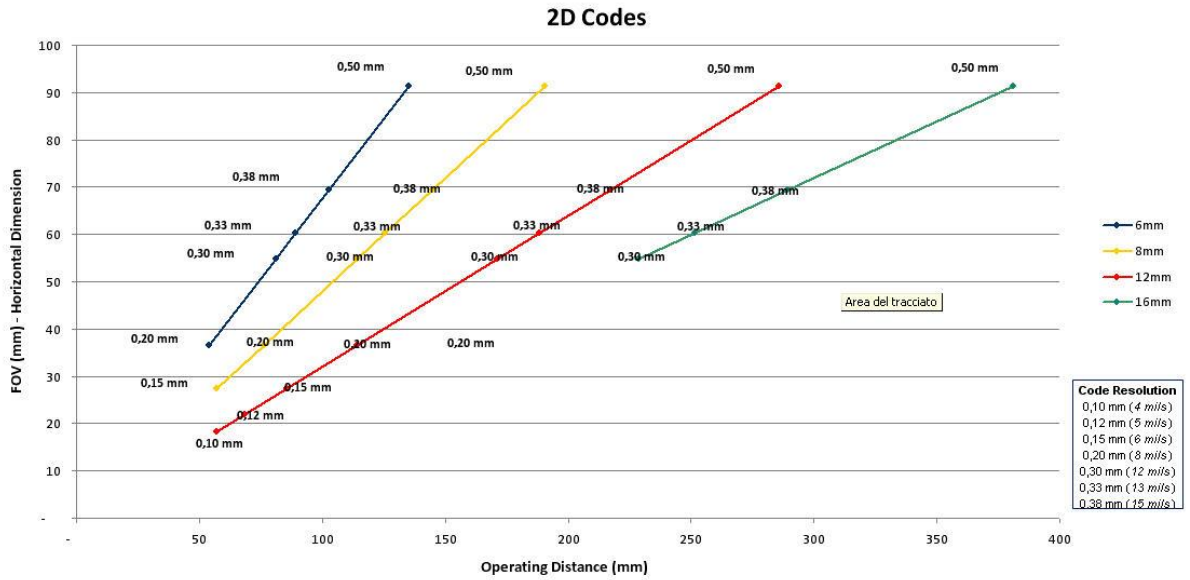
- **Size**: the number of rows and columns composing a Datamatrix symbol that can be decoded using DataVS2 can vary from 8x8 to 144x144. All possible value ranges are accepted for a Datamatrix symbol.
- **Coding**: no limits are foreseen relative to the type of coding used for the data inside the symbol. In the result string sent by DataVS2 on the Ethernet and Serial channels, the code read is represented in a byte vector format; the byte decoding is made by the client that reads it.

NOTE

DataVS2 GUI decodes all bytes of the code read according to the ASCII table. The contents of a Datamatrix symbol that has not been coded in an ASCII format cannot be displayed.

- **Resolution**: the number of pixels per symbol module to decode can vary in the [3, 20] range. This value represents the resolution that characterizes the symbol to decode inside an image.

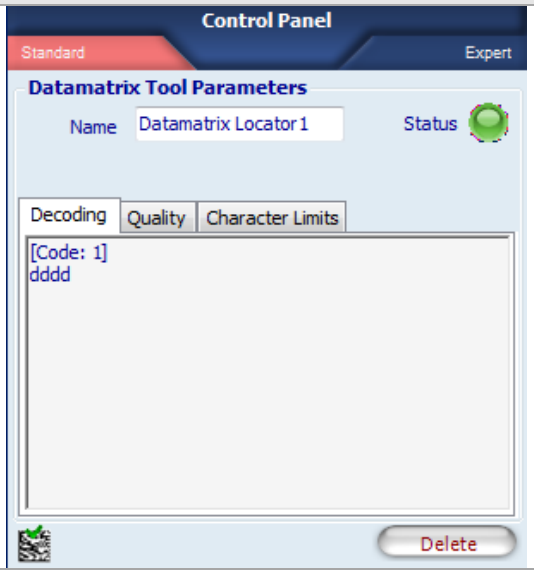
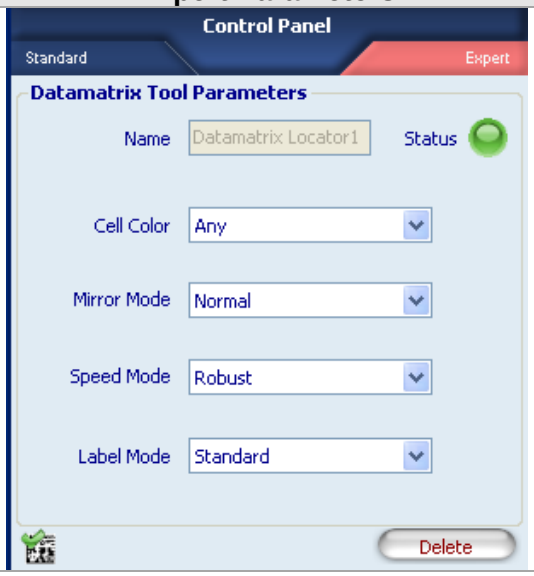
  Please refer to the following graphic for more information on the resolution of the code to read, expressed in mm/FOV:



- **Orientation Angle**: the symbol can have all the possible orientation angles in the ROI.
- **Edges (Quiet Zone):** Datamatrix symbols having Edge dimensions equal to 1 module (the dimension of a unit containing information inside the symbol) are accepted.

Datamatrix Locator Parameters

The following tables show the Datamatrix Locator parameters:

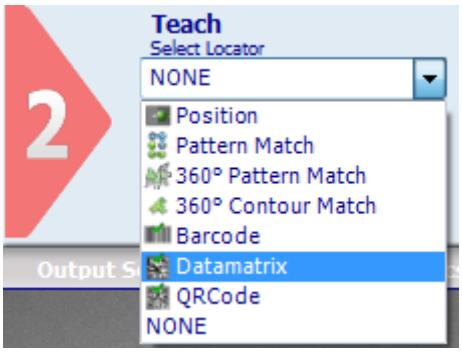
Standard Parameters	Expert Parameters
	

Parameters	Description	Type	Default
Name	The Name assigned to the Locator tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	Datamatrix Locator
Status	Tool that indicates a result: • Green: SUCCESS. the Datamatrix Locator has decoded a Datamatrix symbol in the ROI. • Red: FAILURE. the Datamatrix Locator hasn't decoded any symbol in the ROI or the decoded symbol is incompatible with the filters set.	Standard, Output	
Decoding	The **Decoding** field displays the contents of the decoded symbol.	Standard, Output	
Quality	These values represent the general quality of the decoded symbol measured by the Datamatrix Locator. The quality of a Datamatrix symbol is represented by the measurement of the following parameters, expressed according to the [0, 1, 2, 3, 4] value scale, where 4 corresponds to the maximum quality value of a code: • **Symbol Contrast**: represents the symbol contrast level expressed as the difference of reflectivity between light and dark parts of the symbol. • **Axial Non Uniformity**: represents the average space difference between two sides of the symbol arranged along the X and Y axes. • **Decode Grade**: represents the result supplied by the reference decoding algorithm used by the Datamatrix Locator. • **Overall Grade**: is equal to the minimum conformity value detected in the quality parameters described above. This value represents the general quality of the symbol and is also graphically displayed next to the **Calculated** label. • The **Expected Value** parameter of the Quality field can	Standard, Output	

Parameters	Description	Type	Default
	be used to set a filter on the expected Quality value. The Datamatrix Locator, in each execution, verifies that the **Calculated Value** is higher or equal to the **Expected Value**.		
Character Limits	It is possible to insert a filter on the number of read characters. The codes which have a number of characters greater than the maximum or less than the minimum are not returned by the tool.	Standard, Input	[0, 3116]
Cell Colour	Represents the colour of the Datamatrix symbol to decode: • **Black on White**: black on white background. • **White on Black**: white on black background. • **Any**: the Locator in this case tries to decode the symbol according to both the previous options.	Expert, Input	Any
Mirror Mode	Defines the orientation of the cells with respect to the **Finder Pattern** of the symbol. The codes can be Normal or Mirrored.	Expert, Input	Normal
Speed Mode	Influences the accuracy level used by the Datamatrix Locator during the search phase; the higher the accuracy value, the more precise the calculation of the similarity degree of the object found. Low levels of Accuracy increase the searching process speed: we suggest to use low Accuracy values when the execution time does not respect the requirements.	Expert, Input	Robust
Label Mode	Defines the print typology of the symbol, that has to be found among the following alternatives: • **Standard**: inkjet printing; in this case the symbol presents high contrast. • **Dotpeen**: direct marking print. • **Low Contrast**: inkjet or direct marking print characterized by a low contrast between symbol colour and background.	Expert, Input	Standard
Delete	Deletes the Locator from the current Inspection.		

Example

- Add a Datamatrix Locator to the current Inspection.
- Select “Datamatrix” from the drop-down menu of the available Locators.



- Position the Locator on the reference image in the IMAGE PANEL: the pointer changes shape as soon as it enters in the work area. Left-click once on any position to create a default rectangular ROI.
- At the same time, the tool characteristics and the result of its application on the reference image appear in the control panel.
- Assure that the reference image contains the Datamatrix symbol to decode and that it corresponds to the characteristics specified above.
- The result indicated by STATUS is also highlighted by the ROI border changing colour; the result will be SUCCESS if the symbol is decoded. If the Locator FAILS, check that the ROI has been correctly positioned and that the symbol is clearly visible inside the image, possibly choosing a better reference image of the part to inspect.

- It is then possible to test the Inspection configured with images containing different symbols.

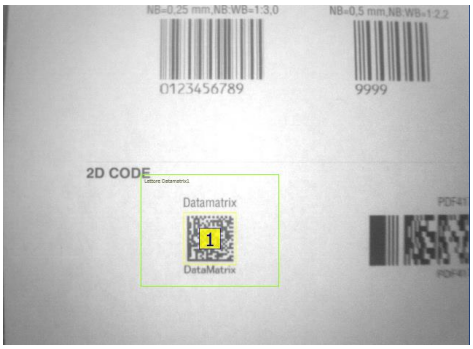


Fig. A: SUCCESS

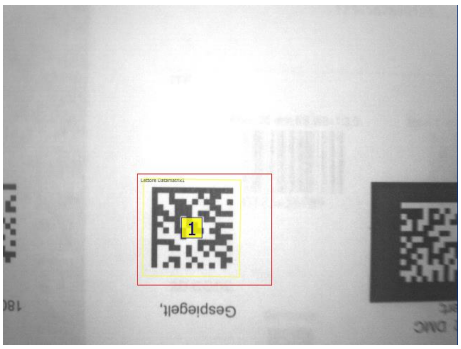


Fig. B: FAILURE

5.3.8 Barcode Locator

(only available in DataVS2 PRO Model)

The Barcode Locator is able to detect the position and the orientation of one Barcode symbol inside a well specified area of the image.

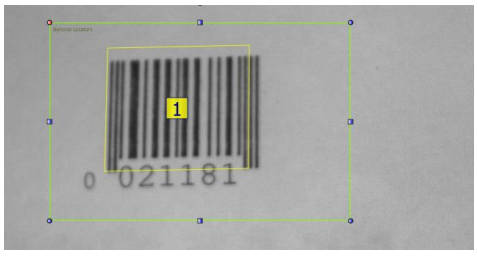
Barcode Locator: selection and positioning

You can select the Barcode Locator from the drop-down Locators menu in STEP2. The Barcode Locator is identified by the following icon:

When you are in STEP 2 the Barcode Locator allows you to make the following changes:

- ROI Moving: the position of the ROI can be changed in the Reference Image.
- ROI Resizing: the width and the height of the ROI can be changed in the Reference Image.
- ROI Rotation: the orientation angle of the ROI can be changed in the Reference Image.

The symbol found is graphically represented inside the ROI with a yellow frame that identifies the vertexes and a rectangle containing the index of the code found, visualized in the centre of the code itself.



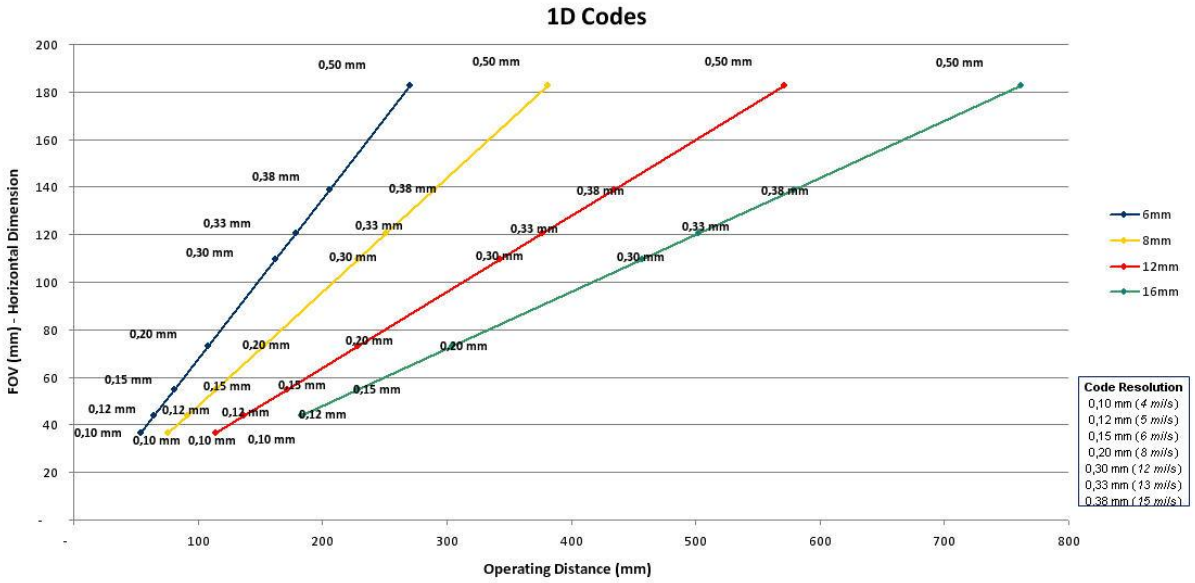
Barcode Symbol Characteristics

The DataVS2 Barcode Locator decodes the Barcode symbol with the following characteristics:

- **Coding**: no limits are foreseen relative to the type of coding used for the data inside the symbol. In the result string sent by DataVS2 on the Ethernet and Serial channels, the code read is represented in a byte vector format; the byte decoding is made by the client that reads it.

NOTE	*DataVS2 GUI decodes all bytes of the code read according to the ASCII table. The contents of a Barcode symbol that has not been coded in an ASCII format cannot be displayed.*

- **Resolution**: the following graphic provides more information on the resolution of the code to read, expressed in mm/FOV:

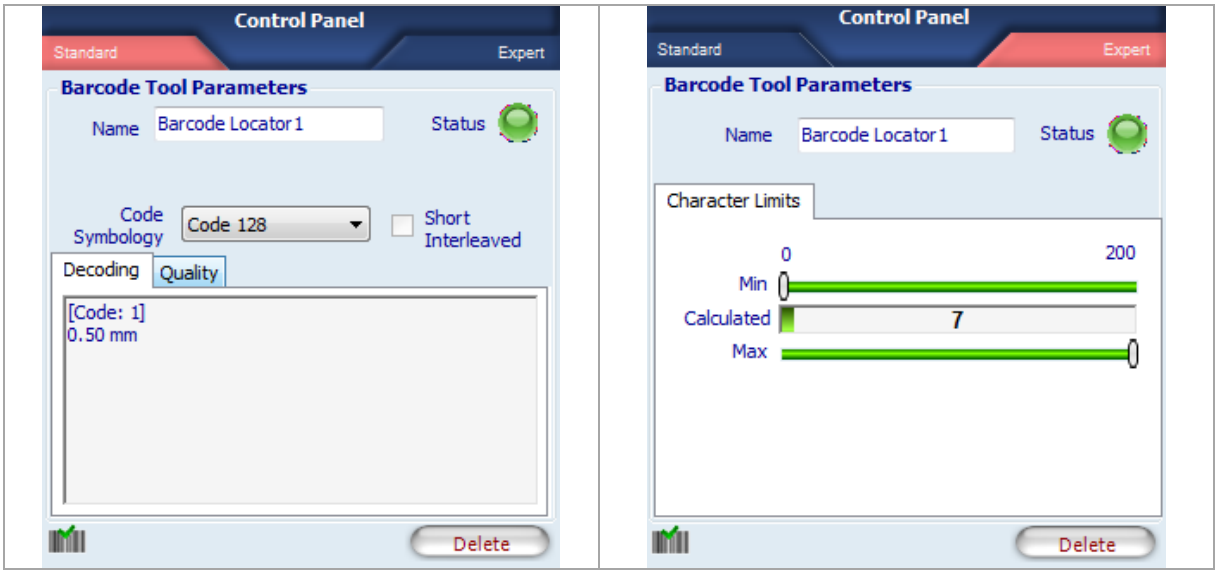


- **Orientation Angle**: the symbol can have all the possible orientation angles in the ROI.
- **Edges (Quiet Zone):** Barcode symbols having Edge dimensions equal to 1 module (the dimension of a unit containing information inside the symbol) are accepted.

Barcode Locator Parameters

The following table shows the Barcode Locator parameters:

Standard Parameters	Expert Parameters



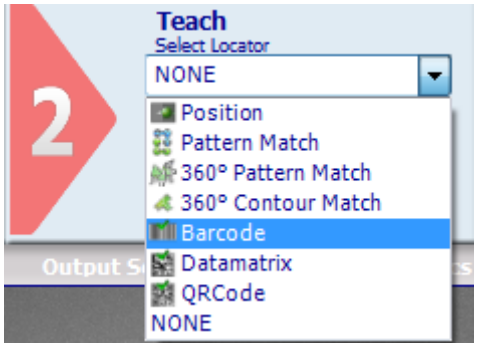
Parameters	Description	Type	Default
Name	The Name assigned to the Control tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	Barcode
Status	Tool that indicates a result: • Green: SUCCESS. The Barcode Locator has decoded Barcode symbol in the ROI. • Red: FAILURE. the Barcode Locator hasn't decoded any symbol in the ROI or the decoded symbol is incompatible with the filters set.	Standard, Output	
Decoding	The **Decoding** field displays the contents of the decoded symbol.	Standard, Output	
Code Symbology	The Code Symbology field defines the type of Barcode code searched for in the ROI. When no value is selected, the Locator automatically sets the Symbology retrieving it from the code read in the Teach phase (Step 2). The possible symbologies accepted for a Barcode that can be decoded with DataVS2 PRO are given below: • Code 128 • Code 39 • Codabar • Interleaved • EAN 13 • EAN 8 • UPC E • Postnet • IMB • Pharmacode (cannot be set automatically) • PDF 417 • UPC A • GS1-128 • Code 93	Standard, Input	Auto
Short Interleaved	It is possible to enable this parameter only if it's selected the Interleaved symbology. If enabled, you can read codes with only two characters, otherwise the minimum is five characters.	Standard, Input	Disabled

Parameters	Description	Type	Default
Quality	These values represent the general quality of the decoded symbol measured by the Barcode Locator. The quality of a Barcode symbol is represented by the measurement of the following parameters, expressed according to the [0, 1, 2, 3, 4] value scale, where 4 corresponds to the maximum quality value of a code: (*) • **Symbol Contrast**: the symbol contrast is the ratio between the lowest reflectivity of a bar and the highest reflectivity of a space. The higher the contrast, the better the symbol quality. • **Minimum Edge Contrast**: represents the minimum reflectivity. The lower this value (dark bars), the better the contrast. • **Modulation**: modulation is the ratio between Minimum Edge Contrast and Symbol Contrast. • **Decodability**: represents the result sent by the reference decoding algorithm used by the Barcode Locator. The decoding is a prerequisite for the symbol quality check. • **Overall Grade**: is equal to the minimum conformity value detected in the quality parameters described above. This value represents the general quality of the symbol and is also graphically displayed next to the **Calculated** label. • The **Expected Value** parameter of the Quality field can be used to set a filter on the expected Quality value. The Barcode Locator, in each execution, verifies that the **Calculated Value** is higher or equal to the **Expected Value**.	Expert, Output	
Character Limits	Allows specifying the minimum and maximum number of characters that have to be present in the symbol for the tool result to be SUCCESS. • **Calculated**: represents the number of characters decoded in the symbol. • **Minimum**: represents the minimum number of characters that have to be decoded in the symbol for the Locator result to be SUCCESS. • **Maximum**: represents the maximum number of characters that have to be decoded in the symbol for the Locator result to be SUCCESS.		
Delete	Deletes the Locator from the current Inspection.		

(*) Exceptions are the following symbology: IMB, Postnet, PDF417 and Pharmacode. For these symbology all parameters are not calculated.

Example

- Add a Barcode Locator to the current Inspection.
- Select “Barcode” from the drop-down menu of the available Locators.



- Position the Locator on the reference image in the IMAGE PANEL: the pointer changes shape as soon as it enters in the work area. Left-click once on any position to create a default rectangular ROI.

- At the same time, the tool characteristics and the result of its application on the reference image appear in the control panel.
- Assure that the reference image contains the Barcode symbol to decode and that it corresponds to the characteristics specified above.
- The result indicated by STATUS is also highlighted by the ROI border changing colour; the result will be SUCCESS if the symbol is decoded. If the Locator FAILS, check that the ROI has been correctly positioned and that the symbol is clearly visible inside the image, possibly choosing a better reference image of the part to inspect.
- It is then possible to test the Inspection configured with images containing different symbols.

Fig. A: SUCCESS

Fig. B: FAILURE

5.3.9 QRCode Locator

(only available in DataVS2 PRO Model)

The QRCode Locator is able to detect the position and the orientation of one QR code symbol inside a well specified area of the image.

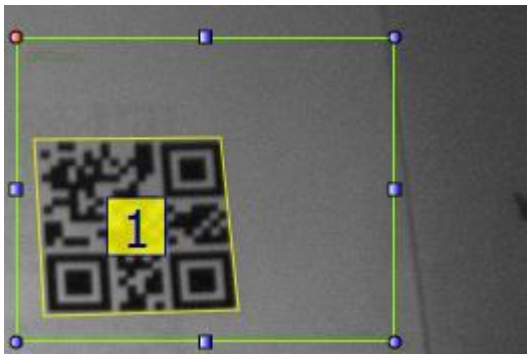
QRCode Locator: selection and positioning

You can select the QRCode Locator from the drop-down Locators menu in STEP2. The QRCode Locator is identified by the following icon:

When you are in STEP 2 the QRCode Locator allows you to make the following changes to the ROI:

- ROI Moving: the position of the ROI can be changed in the Reference Image.
- ROI Resizing: the width and the height of the ROI can be changed in the Reference Image.

The symbol found is graphically represented inside the ROI with a yellow frame that identifies the vertexes and a rectangle containing the index of the code found, visualized in the centre of the code itself.



QRCode Symbol Characteristics

The DataVS2 QRCode Locator decodes the QRCode symbol with the following characteristics:

- **Coding**: no limits are foreseen relative to the type of coding used for the data inside the symbol. It is possible to read QR codes (model 1 and model 2) and Micro QR codes. In the result string sent by DataVS2 on the Ethernet and Serial channels, the code read is represented in a byte vector format; the byte decoding is made by the client that reads it.

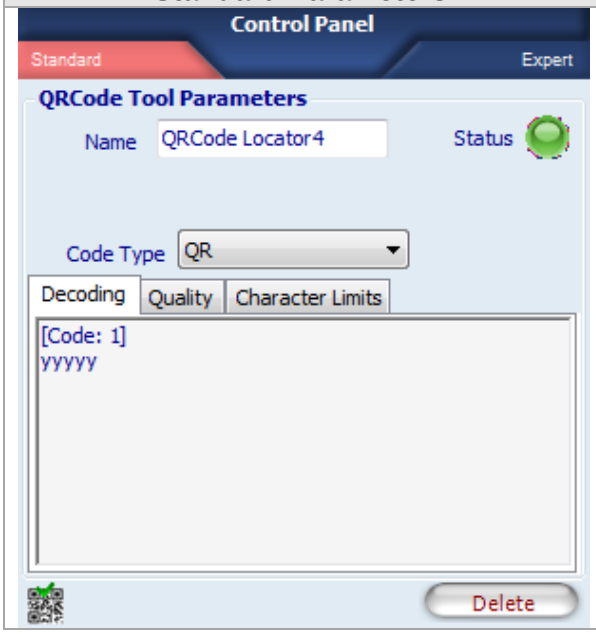
NOTE

DataVS2 GUI decodes all bytes of the code read according to the ASCII table. The contents of a QR symbol that has not been coded in an ASCII format cannot be displayed.

- **Size**: the number of rows and columns composing a QR Code Model 1 symbol that can be decoded using DataVS2 can vary from 21x21 to 73x73, for QR Code Model 2 it can vary from 21x21 to 177x177, increasing in both cases in steps of four modules per side. For the Micro QR symbol instead the number of rows can vary from 11x11 to 17x17, increasing in both cases in steps of two modules per side.
- **Resolution**: the number of pixels per symbol module to decode can vary in the [4, 20] range. This value represents the resolution that characterizes the symbol to decode inside an image.
- **Orientation Angle**: the symbol can have all the possible orientation angles in the ROI.
- **Edges (Quiet Zone):** QR symbols having Edge dimensions equal to 1 module (the dimension of a unit containing information inside the symbol) are accepted.

QRCode Locator Parameters

The following tables show the QRCode Locator parameters:

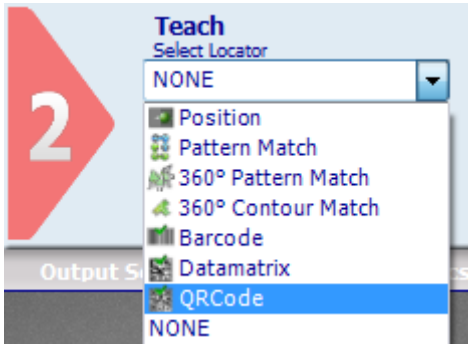
Standard Parameters	Expert Parameters
	

Parameters	Description	Type	Default
Name	The Name assigned to the Locator tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	QRCodeLocator
Status	Tool that indicates a result: • Green: SUCCESS. the QRCode Locator has decoded a QR symbol in the ROI. • Red: FAILURE. the QRCode Locator hasn't decoded any symbol in the ROI or the decoded symbol is incompatible with the filters set.	Standard, Output	
Code Type	The Code Type field defines the type of QR code searched in the ROI. It could be QR or Micro QR	Standard, Input	QR
Decoding	The **Decoding** field displays the contents of the decoded symbol.	Standard, Output	
Quality	These values represent the general quality of the decoded symbol measured by the QRCode Locator. The quality of a QR symbol is represented by the measurement of the following parameters, expressed according to the [0, 1, 2, 3, 4] value scale, where 4 corresponds to the maximum quality value of a code: • **Symbol Contrast**: represents the symbol contrast level expressed as the difference of reflectivity between light and dark parts of the symbol. • **Axial Non Uniformity**: represents the average space difference between two sides of the symbol arranged along the X and Y axes. • **Decode Grade**: represents the result supplied by the reference decoding algorithm used by the QRCode Locator. • **Overall Grade**: is equal to the minimum conformity value detected in the quality parameters described	Standard, Output	

Parameters	Description	Type	Default
	above. This value represents the general quality of the symbol and is also graphically displayed next to the **Calculated** label. • The **Expected Value** parameter of the Quality field can be used to set a filter on the expected Quality value. The QRCode Locator, in each execution, verifies that the **Calculated Value** is higher or equal to the **Expected Value**.		
Character Limits	It is possible to insert a filter on the number of read characters. The codes which have a number of characters greater than the maximum or less than the minimum are not returned by the tool.	Standard, Input	[0, 3116]
Cell Colour	Represents the colour of the QR symbol to decode: • **Black on White**: black on white background. • **White on Black**: white on black background. • **Any**: the Locator in this case tries to decode the symbol according to both the previous options.	Expert, Input	Any
Mirror Mode	Defines the orientation of the cells with respect to the **Finder Pattern** of the symbol. The codes can be Normal or Mirrored.	Expert, Input	Normal
Delete	Deletes the Locator from the current Inspection.		

Example

- Add a QRCode Locator to the current Inspection.
- Select “QRCode” from the drop-down menu of the available Locators.



- Position the Locator on the reference image in the IMAGE PANEL: the pointer changes shape as soon as it enters in the work area. Left-click once on any position to create a default rectangular ROI.
- At the same time, the tool characteristics and the result of its application on the reference image appear in the control panel.
- Assure that the reference image contains the QR symbol to decode and that it corresponds to the characteristics specified above.
- The result indicated by STATUS is also highlighted by the ROI border changing colour; the result will be SUCCESS if the symbol is decoded. If the Locator FAILS, check that the ROI has been correctly positioned and that the symbol is clearly visible inside the image, possibly choosing a better reference image of the part to inspect.
- It is then possible to test the Inspection configured with images containing different symbols.

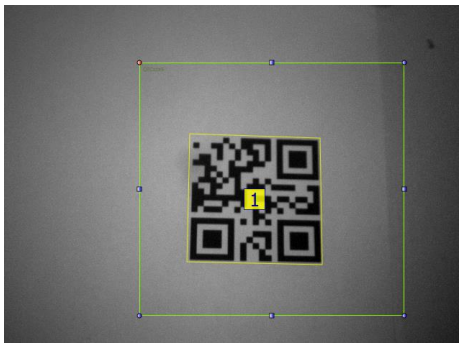


Fig. A: SUCCESS

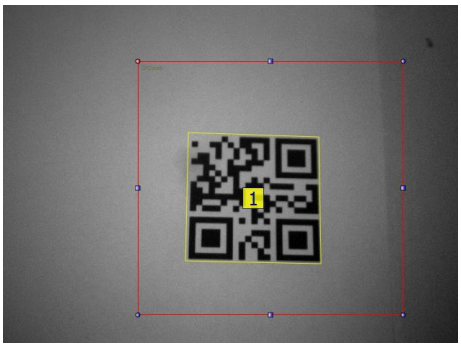


Fig. B: FAILURE

5.4 CONTROLS

Controls are used to verify certain characteristics in defined regions (ROI) of the acquired images according to a specific Image Processing technique.

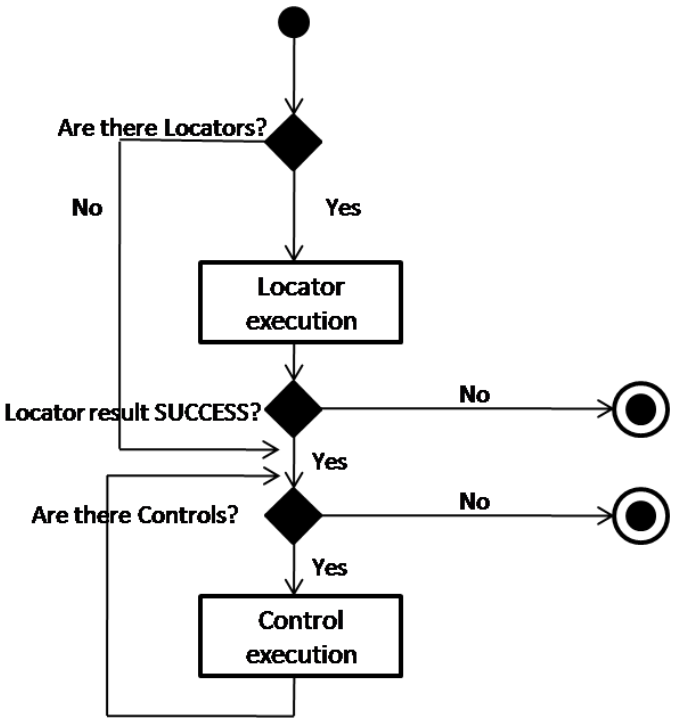
DataVS2 GUI offers you the possibility to graphically configure Controls and Locators on the IMAGE PANEL using the mouse. The Controls setup is done in STEP 2.

More than one control can be added in the current inspection. The number of Controls to include depends on the application needs.

When multiple Controls are present in the same Inspection they are executed one-by-one, according to the insertion order.

If a Locator is present in the same Inspection, this will be carried out first. If the result of the Locator corresponds to a SUCCESS, all the controls are carried out, otherwise none.

The following image represents with a flowchart the Image Processing phase steps.



5.4.1 Selection and Positioning

Depending on the DataVS2 model, (see the table in par. 5.2.2), several types of Controls may be selected from the pull-down menu under the “Select Control” label in STEP 2:

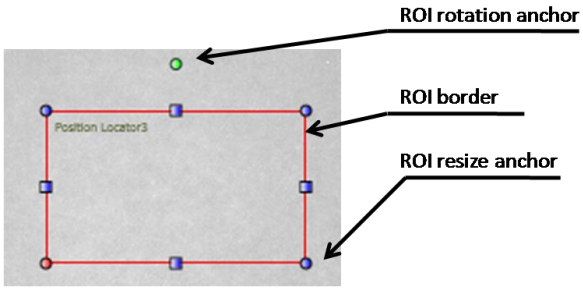
- Brightness
- Contrast
- Contour Match
- Edge Count
- Width
- Pattern Match
- Position

- OCV
- Datamatrix
- Barcode
- QRCode
- Contour Counter 360°
- Defect Finder 360°

NOTE

If you move your mouse cursor on the IMAGE PANEL after you have selected a Control it will change appearance. It will assume the shape of the relative Control icon, as shown in the pull-down menu.

After selecting the Control, move the mouse in the IMAGE PANEL. Now click on any position on the image. This will create a ROI frame as shown below.



If you want to change the ROI size, left-click and drag one of the blue anchors you find on the ROI borders. If you want to move the ROI, place the cursor over one of the 4 ROI borders; it will change its icon into a 4 crossed-arrowed one. Press and hold the left mouse button and drag the rectangle to the desired position of the image.
If you want to rotate the ROI, left-click and drag the green anchor on the top of the ROI: the ROI will rotate around its barycentre.

After adding the Control to the current Inspection, the DataVS2 GUI CONTROL PANEL will display its parameters.

NOTE

Not all Tool allow to carry out all the modifications on the relative ROIs (ROI Moving, ROI Resizing, ROI Rotation).

NOTE

Controls cannot compensate position offsets of the object in the image. Once defined the ROI location and dimension in the Reference Image, the Control will always process the position of the current acquired image corresponding to the ROI. You have to use a Locator if the position of the feature to control (i.e. the position of the ROI) or if the object can move within the field of view.

5.4.2 Brightness

The Brightness Control calculates the average brightness of the pixels included in the ROI. The brighter the pixels, the higher the calculated value.

The Brightness Control can be used to verify if a certain feature is present or absent in the portion of the image included in the ROI borders.
Examples of usage for Brightness are:

- Checking the presence of a white label on a dark box;
- Differentiating dark objects from white objects.

Brightness selection and positioning

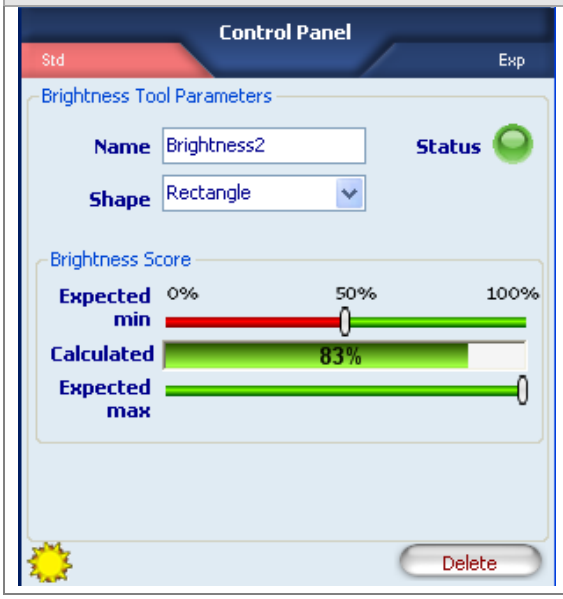
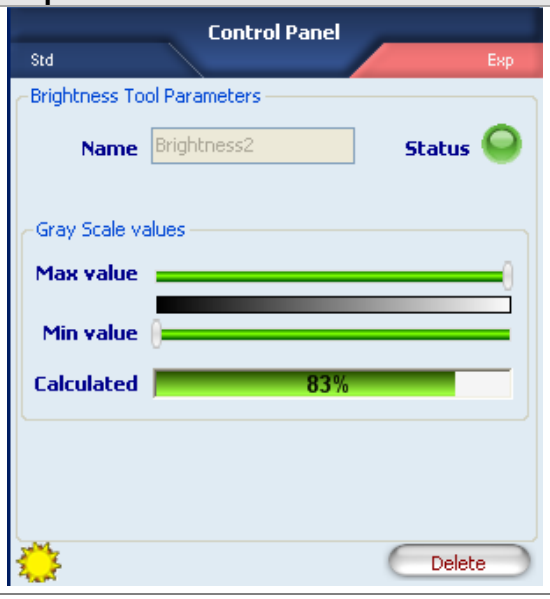
You can select the Brightness Control from the Controls pull-down menu in STEP2. The Brightness Control is identified by the following icon:

When you are in STEP 2, the Contrast Control ROI allows the following variations:

- ROI Moving: the position of the ROI can be changed in the Reference Image.
- ROI resizing: the width and the height of the ROI can be changed in the Reference Image.
- ROI Rotation: the orientation of the ROI can be changed in the Reference Image (only in the DataVS2 GUI 1.2 version).
- Change the shape of the ROI: the ROI shape can be changed from rectangular to circular and vice versa (only in the DataVS2 GUI 1.2 version).

The average brightness calculated for the pixels included in the ROI edges is shown in a percentage value in the CONTROL PANEL. If the value calculated is inside the limits defined in the interval of the expected values (Minimum expected value, Maximum expected value) the control result will be positive (SUCCESS), otherwise FAIL.
If SUCCESS, the Brightness ROI background is green, otherwise it is red.

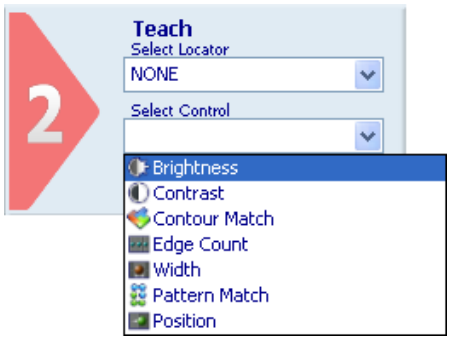
Brightness parameters
The following image and the table display the Brightness parameters.

Standard Parameters	Expert Parameters
	

Parameters	Description	Type	Default
Name	The Name assigned to the Control Tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	Brightness
Status	A tool result indicator . • Green: SUCCESS. The Brightness has found the target pattern with a score higher that the expected one. • Red: FAILURE. Brightness hasn't found the target pattern or the score of the found pattern is lower than the expected one.	Standard, Output	
Shape	Represents the current shape of the ROI. Two are the possible values: • Rectangular • Elliptical Adjust the shape in order to better fit the characteristics of the region of the image to inspect.	Standard, Input	Rectangular
Calculated	The score representing the calculated brightness average of the pixels included in the ROI.	Standard, Output	
Minimum	The minimum value that the calculated brightness average must have in order to make the result be SUCCESS. All calculated values lower than the expected Min will make FAIL the Brightness result	Standard, Input	50%
Max	The maximum value that the calculated brightness average must have in order to make the result be SUCCESS. All calculated values higher than the expected Max will make FAIL the Brightness result. ***Note:*** this parameter is available from the DataVS2 GUI 1.2 version onwards. The parameter is disabled for the Inspection versions previous to the 2.1. (you can check the current Inspection version in the INSPECTION EXPLORER)	Standard, Input	100%
Delete	Deletes the Control from the current Inspection		
Gray scale: Max	The maximum gray scale value that will be used for brightness average calculation. All pixels having gray scale values higher than the Maximum expected will be ignored for the average calculation.	Expert, Input	255
Gray scale: Minimum	The minimum gray scale value that will be used for brightness average calculation. All pixels having gray scale values lower than the Minimum expected will be ignored for the average calculation.	Expert, Input	0

Example

- Add "Brightness" control with rectangular ROI to current inspection.
- Select "Brightness" from the pull-down menu:



- Position the control on the reference image in the work area: the pointer changes shape as soon as it enters in the work area. Left-click once at any position to create a default rectangular ROI. At the same time, the control features and the result of its application to the reference image appear in the control panel.
- Set ROI size and position as required: in this example, the ROI shape is not changed. Set ROI position and size as explained previously.
- Adjust Expected Min and Expected max thresholds in order to fit your needs.
- The result indicated by STATUS is also highlighted by the ROI border changing colour.
- As soon as the values change, result is calculated without delay.
- Hereinafter, the Figure A and B show a working example of the Brightness Control. In this particular case, the Brightness Control verifies the presence of pills inside the blister cells.

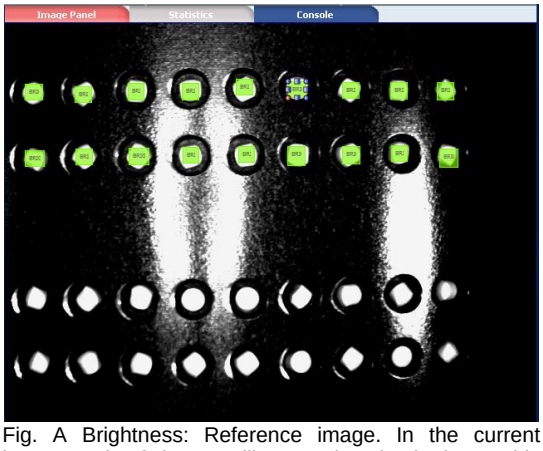


Fig. A Brightness: Reference image. In the current image each of the 19 pills must be checked. For this reason, 19 Brightness Controls have been used. Each Control verifies the presence of a pill.

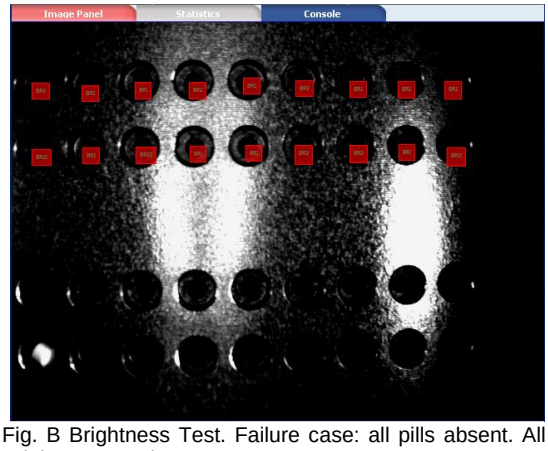


Fig. B Brightness Test. Failure case: all pills absent. All the Brightness results are FAILURE.

5.4.3 Contrast

The Contrast Control calculates the difference, given as ratio, between the lightest and the darkest group of pixels within the ROI.
Considering a ROI characterized by 50% of black pixels and the remaining pixels white, the calculated contrast value is 100%. Considering a ROI characterized by pixels having the same brightness level, the calculated contrast value is 0%.

The Contrast Control can be used in the following applications:

- Presence and absence verification of a feature inside the ROI;
- Integrity Control: verify if the assembled object contains all the requested parts.

Contrast selection and positioning

You can select the Contrast Control from the Controls pull-down menu in STEP2. The Contrast Control is identified by the following icon:

When you are in STEP 2, the Contrast Control ROI allows the following variations:

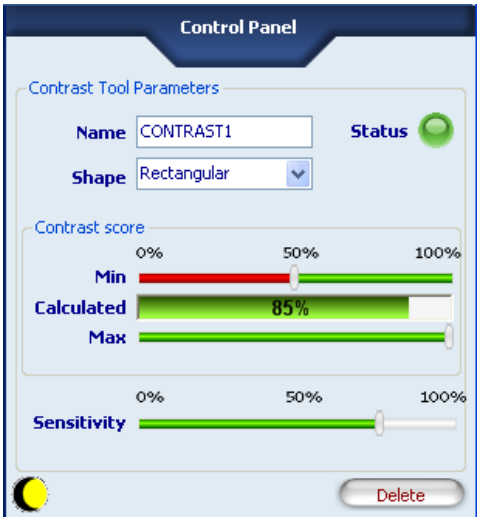
- ROI Moving: the position of the ROI can be changed in the Reference Image.
- ROI resizing: the width and the height of the ROI can be changed in the Reference Image.
- ROI Rotation: the orientation of the ROI can be changed in the Reference Image (only in the DataVS2 GUI 1.2 version).

Change the shape of the ROI: the ROI shape can be changed from rectangular to circular and vice versa (only in the DataVS2 GUI 1.2 version).

The Contrast value calculated in image part delimited in the ROI is shown as a percentage value in the CONTROL PANEL. If the value calculated is inside the limits defined in the interval of the expected values (Minimum, Maximum) the control result will be positive (SUCCESS), otherwise FAIL.
If SUCCESS the Brightness ROI background is green, otherwise it is red.

Contrast parameters

The following image and the table display the Contrast parameters.



Parameters	Description	Type	Default
Name	The Name assigned to the Control Tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	Contrast
Shape	Represents the current shape of the ROI. Two are the possible values: Rectangular Elliptical Adjust the shape in order to better fit the characteristics of the region of the image to inspect.	Standard, Input	Rectangular
Status	A tool result indicator . **Green**: SUCCESS. The Contrast has found the target pattern with a score higher that the expected one. **Red**: FAILURE. Contrast hasn't found the target pattern or the score of the found pattern is lower than the expected one.	Standard, Output	
Calculated	The score representing the calculated contrast value of the pixels included in the ROI.	Standard, Output	

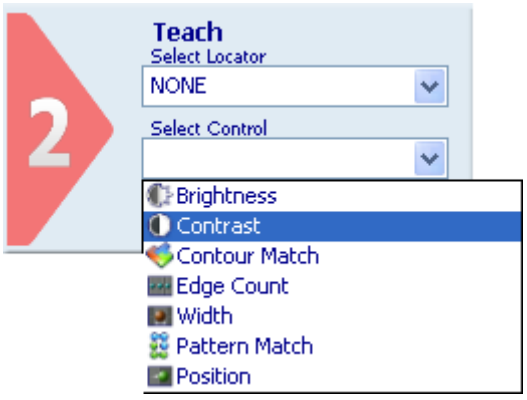
Parameters	Description	Type	Default
Minimum	The minimum value that the calculated contrast value must have in order to make the result be SUCCESS. All calculated values lower than the expected Min will produce FAIL as the Contrast result	Standard, Input	50%
Max	The maximum value that the calculated contrast value must have in order to make the result be SUCCESS. All calculated values higher than the expected Max will produce FAIL as the Contrast result. ***Note:*** this parameter is available from the DataVS2 GUI 1.2 version onwards. The parameter is disabled for the Inspection versions previous to the 2.1. (you can check the current Inspection version in the INSPECTION EXPLORER)	Standard, Input	100%
Delete	Deletes the Control from the current Inspection		
Sensitivity	The scale factor used in calculating the contrast value; useful to highlight the value calculated when the characteristic to detect presents a low contrast between the gray pixel levels. 0% the contrast value will be lowered 50% the contrast value will be left as it is 100% the contrast value will be increased	Standard, Input	50%

NOTA

The Contrast Tool is often used to check the presence of a characteristic inside the ROI, for example, a character string. Unfortunately, it is quite common that the amount and the gray level of pixels representing the characteristic inside the ROI are not enough to generate a good level of Contrast difference between the good part and the bad part. This means that the calculated value in the good image and in the bad image do not differ and consequently, the application could not be performed.
If this is your case, we recommend using the Sensitivity parameter to highlight the differences between the good and the bad parts.

Example

- Add "Contrast" control with rectangular ROI to current inspection.
- Select "Contrast" from the pull-down menu:



- Position the control on the reference image in the work area: the pointer changes shape as soon as it enters in the work area. Left-click once on any position to create a default rectangular ROI. At the same time, the control characteristics and the result of its application on the reference image appear in the control panel.
- The result indicated by STATUS is also highlighted by the ROI border changing colour.
- As soon as the values change, result is calculated without delay.
- The control thus added to current inspection is also displayed in the Inspection Explorer: its name is listed under "Tools" (name specified in the Name field). If you expand that menu level (click on symbol ⊞ next to the name), you will see the parameters set previously.
- Hereinafter a possible usage scenario for the Contrast Control Tool is presented. In this particular inspection, the presence of a label on a box has to be checked. If the label is correctly placed, the overall contrast of the ROI increases its value, if the printing is not present the contrast value should be very low (the contrast of a uniform area is null). The Fig. A shows a successful check, the Fig. B on the contrary shows a failure.

Fig. A: SUCCESS

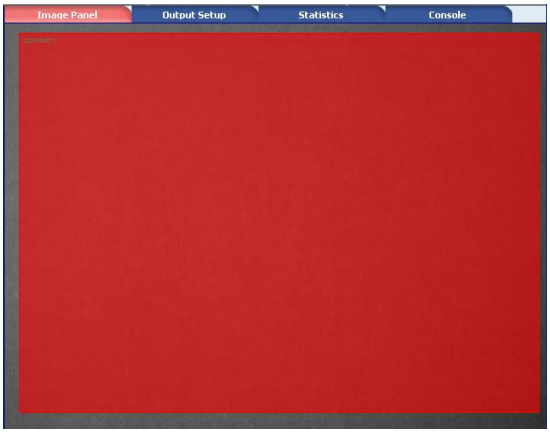


Fig. B: FAILURE

5.4.4 Contour Match

The Contour Match Control calculates the value relative to the comparison between the contour of a target object and the contour of an object included in the current image. This verifies how similar the shapes of the current object and target object are, returning a percentage value that can vary from 0% to 100% (Contour match ►).

As done by all other controls, Contour Match works in a well-defined Region of Interest.
Possible uses are:

- Presence and absence verification of a characteristic inside the ROI;
- Integrity Control: verify if the assembled object contains all the requested parts.

NOTE

Contour Match searches for Closed Contours only. A contour is closed if it is completely included inside ROI borders and if the region of the pixels belonging to the contour defines a closed shape (the first contour point is identical to the last contour point). For example, a circle or a square completely inside the ROI both comply to these requirements.

NOTE

Contour Match can compensate any rotation displacements of the object up to 360 degrees if the characteristic to control is completely inside the ROI.

If the contour is not completely included inside the ROI, ROI borders will be considered as the found Contour.

Contour Match selection and positioning

You can select the Contour Match Control from the Controls pull-down menu in STEP2. The Contour Match Control is identified by the following icon:

When you are in STEP 2 the Contour Match Control ROI allows for following variations:

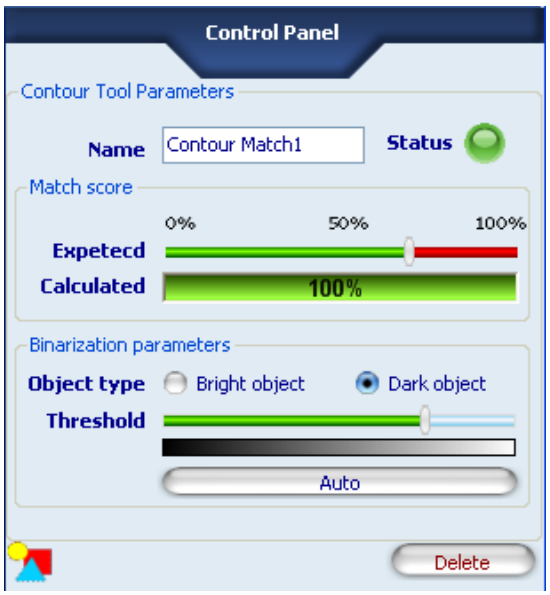
- ROI Moving: the position of the ROI can be changed in the Reference Image.
- ROI resizing: the width and the height of the ROI can be changed in the Reference Image.
- ROI Rotation: the orientation of the ROI can be changed in the Reference Image (only in the DataVS2 GUI 1.2 version).

After an initial setup stage designed to set the master contour on the reference image, it's very important that the ROI contains entirely a closed shape. Then, the system will search for this contour through 360° in target images as they are acquired.

The Contour Match value calculated in the image part delimited inside the ROI is shown as a percentage value in the CONTROL PANEL. If the value calculated is higher than the Minimum value, the result of the control is positive (SUCCESS), otherwise FAIL.
If SUCCESS, the Contour Match ROI border is green, otherwise it is red.

Contour Match parameters

The following image and the table display the Contour Match parameters.

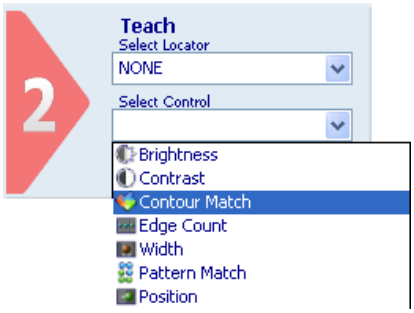


Parameters	Description	Type	Default
Name	The Name assigned to the Control Tool. The maximum length accepted is 256 UTF-8 characters. **Note**: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	Contrast
Status	A tool result indicator . • Green: SUCCESS. The Contour Match has found the target contour with a score higher that the expected one. • Red: FAILURE. Contour Match hasn't found the target contour or the score of the found contour is lower than the expected one.	Standard, Output	
Calculated	The score representing the current degree of similarity between the current detected contour and the target contour.	Standard, Output	
Expected	The expected match score. You may specify of what degree (percent) detected contour must be similar to reference contour in order to be considered valid (100% = perfect match, 0% = no match). This value is set automatically by the system during the first configuration and can be manually adjusted using the slider.	Standard, Input	75%
Object type	Specifies the type of object to find; the possible options are two: • Dark object: searches a dark object on a light background. • Light object: searches a light object on a dark background.	Standard, Input	Dark object
Binarization threshold	Gray scale levels threshold. By changing this value you select which pixels of the ROI are part of the contour and which not. It is compared to the value detected in the acquired image. If this value is below the threshold, the point is not considered part of the object; on the contrary, if the value is over the threshold, the point is considered part of the object. We recommend to set the threshold to a gray	Standard, input	

Parameters	Description	Type	Default
	level value near the average gray level of the background. **Note**: at the first configuration the value is automatically set up by the DataVS2 GUI software. Because it is strictly related to the gray level of the pixels included in the ROI, each time you move the ROI in STEP 2, the binarization threshold is automatically adjusted by the DataVS2 GUI. You can always ask to the DataVS2 GUI to automatically calculate the threshold by pressing the “AUTO” button.		
Auto	Forces the automatic setup of the binarization threshold; it is useful to restore the first configuration after changing the binarization threshold value.	Standard, Input	
Delete	Deletes the Control from the current Inspection		

Example

- Add "Contour Match" control to current inspection to detect the contour of a bright object on a dark background.
- Select "Contour Match" from the pull-down menu:



- Position the control on the reference image in the work area: the pointer changes shape as soon as it enters the work area. Left-click once at any position to create a default rectangular ROI.
- At the same time, the control features and the result of its application to the reference image appear in the control panel.
- Set ROI size and position as required, as explained above.
- The result indicated by STATUS is also highlighted by the ROI border changing colour.
- As soon as values change, result is calculated without delay.
- The object and its contour are highlighted in yellow inside the ROI:
- as can be seen from the example image below, the bright object is highlighted as specified in parameter Object type, which was set to "Bright object". More specifically, the portion of the bright object situated inside the ROI is highlighted.
- Let’s see an application example. In this particular task we want to control the shape of a washer. The Contour Match searches for a closed shape into the ROI area and compares it with the reference shape previously set. Fig. A shows a positive result of the Comparison, Fig. B a negative result: the washer presents 5 points instead of 6.

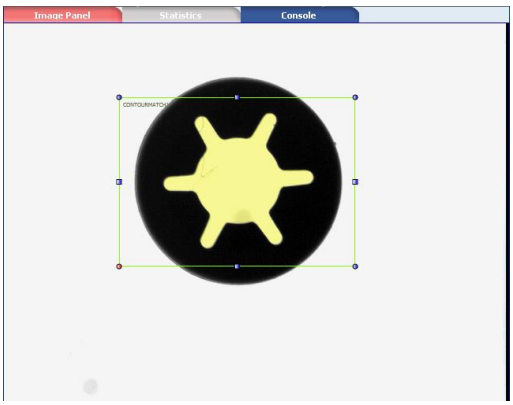


Fig. A: SUCCESS

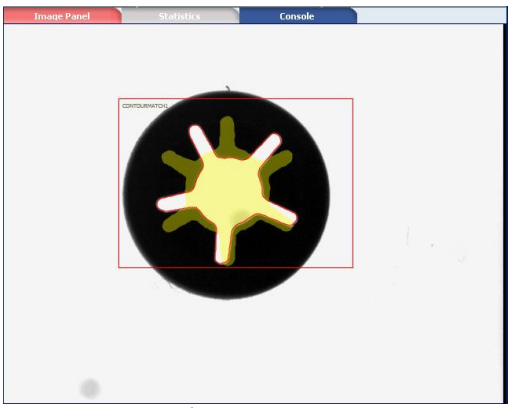


Fig. B: FAILURE

5.4.5 Edge Count

The Edge Count control identifies and counts the number of brightness transitions (so called Edges) located inside the ROI along a line passing through the ROI barycentre.
Based on the first derivative of the pixel brightness variation function, the maximum positive peak is identified if the control is targeting a positive edge, or the maximum negative peak is identified when targeting a negative edge. A point is considered an edge point when its brightness variation exceeds this threshold, which is a function of the peak value found.

Possible uses are:

- Control of part alignment
- Control of absence/presence of threads
- Counting of the number of features

Edge Count selection and positioning

You can select the Edge Count Control from the Controls pull-down menu in STEP2. The Edge Count Control is identified by the following icon:

When you are in STEP 2 the Edge Count Control ROI allows for the following variations:

- ROI Moving: the position of the ROI can be changed in the Reference Image.
- ROI resizing: the width and the height of the ROI can be changed in the Reference Image.
- ROI Rotation: the orientation of the ROI can be changed in the Reference Image (only in the DataVS2 GUI 1.2 version).

The found edges are graphically represented inside the ROI with yellow dots located in the matching point. The searching direction and versus is represented by a yellow arrow passing through the ROI barycentre and going from Left to Right.
If the number of edge found match the expected range the ROI borders will be shown in green, representing a SUCCESS condition.
If no edges are found or if the number of found edges is outside the expected limits the ROI borders will be shown in red, representing a FAILURE condition.

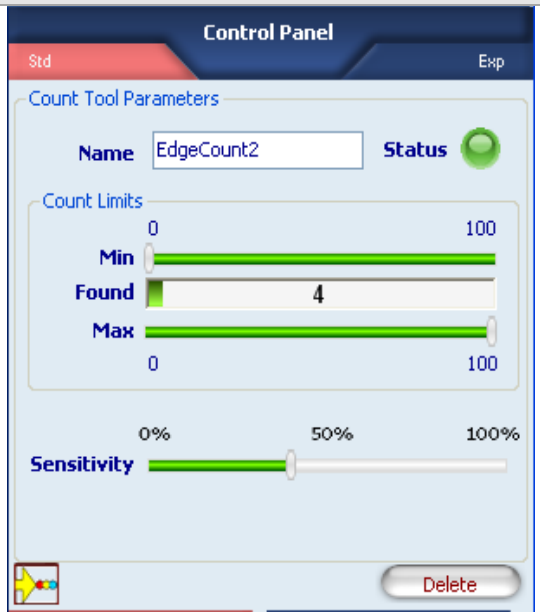
After the Variation Type of the edge (White to Black or Black to White) to be identified has been specified, the Edge Count processes the ROI in the direction shown by the yellow arrow and stops as soon as it comes across the opposite ROI border.

NOTE

The searching direction represented by the yellow line is always referred to the ROI top-left corner. When the ROI orientation is 180 degree you will see the yellow line going from Right to Left.

Edge Count Tool parameters

The following image and the table display the Count Tool parameters.

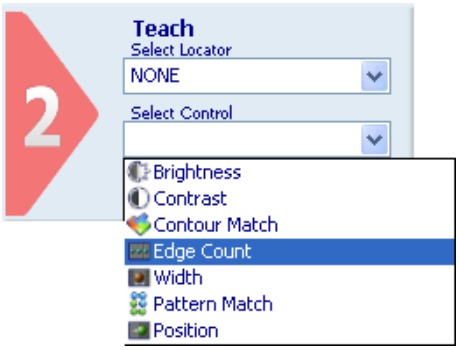
Standard Parameters	Expert Parameters
	

Parameters	Description	Type	Default
Name	The Name assigned to the Control Tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	Edge Count
Status	A tool result indicator . • Green: SUCCESS. The Edge Count has detected an Edge within the range set in “Position limits”. • Red: FAILURE. Edge Count hasn’t found any edge or the found Edge doesn’t match “Position limits”	Standard, Output	
Calculated	Current number of Edges found by the Control inside the ROI along the searching direction.	Standard, Output	
Minimum	Minimum number of expected edges inside the ROI, along the searching direction. This threshold must be compatible with the value found (shown near the "Found" label): this means Minimum must be lower than or equal to "Found”.	Standard, Input	0
Max	Maximum value of expected edges inside the ROI, along the searching direction. This threshold must be compatible with the value found (shown near the "Found" label): this means Max must be greater than or equal to "Found”.	Standard, Input	100
Sensitivity	The degree of accuracy provided when searching for the Edge. The lower the Sensitivity, the greater the accuracy.	Standard, Input	40%
Variation Type	Determines the type of edges to be searched inside the image. Four are the available options: • From white to black: only white to black transitions are detected; • From black to white: only black to white transitions are detected; • Auto: both white to black and black to white transitions are detected; the returned one is the transition with the largest difference between the gray values. • Both: both white to black and black to white transitions	Expert, Input	All

Parameters	Description	Type	Default
	are detected; the returned one is the first transition found.		
Noise Rejection	Reduces the effect that noisy pixels will have on the edge detection. The higher the value, the stronger the brightness variation needed to recognize an edge.	Expert, Input	5
Delete	Deletes the Locator from the current Inspection		

Example

- Add an "Edge count" control to current inspection to identify two edges.
- Select "Edge Count" from the pull-down menu:



- Position the control on the reference image in the work area: the pointer changes shape as soon as it enters in the work area. Left-click once on any position to create a default rectangular ROI.
- At the same time, the control characteristics and the result of its application on the reference image appear in the control panel.
- Set ROI size and position as required, as explained previously.
- The result indicated by STATUS is also highlighted by the ROI border changing colour.
- As soon as the values change, result is calculated without delay.
- Set up the system to recognize all light-to-dark brightness variations as edges. In addition, set the minimum and the maximum number of edges to be detected to '20' (this value can be viewed by placing the mouse pointer on the corresponding bar). In the example on the right, 13 edges with the specified features have been recognized. The control will return a Fail status, identified by the red Status and ROI border.

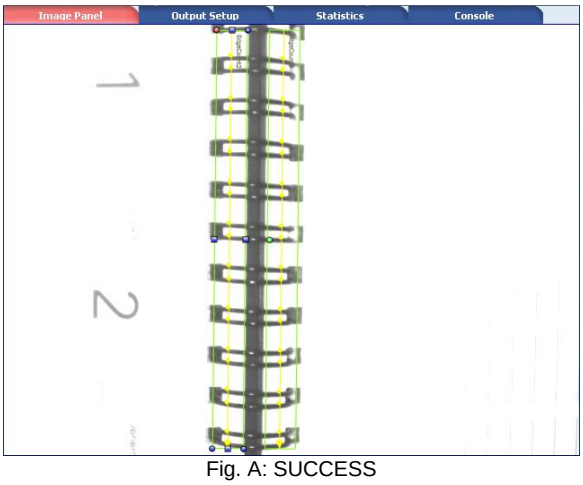


Fig. A: SUCCESS

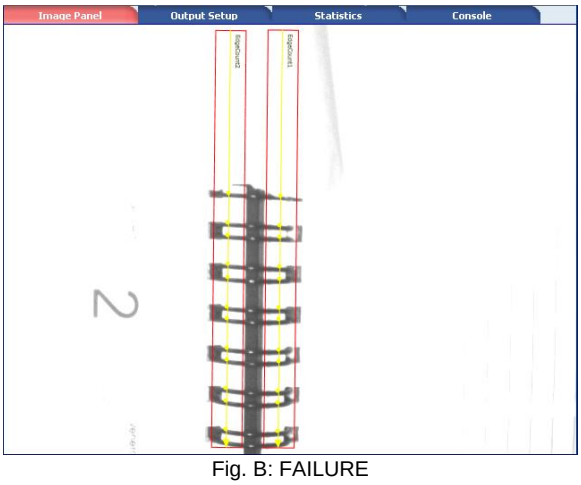


Fig. B: FAILURE

5.4.6 Width

This control is also termed "gauge" and lets you measure the distance between two points (edges).
The calculated value is returned in pixels.
It uses the edge detection technique to identify object edges and then calculates the distance between the edges.

Possible uses are:

- Object width measurement
- Integrity controls: verifying the completeness of an assembled box by measuring its width.

Width: selection and positioning

You can select the Width Control from the Controls pull-down menu in STEP2. The Width Control is identified by the following icon:

When you are in STEP 2 the Edge Count Control ROI allows for the following variations:

- ROI Moving: the position of the ROI can be changed inside the Reference Image.
- ROI resizing: the width and the height of the ROI can be changed inside the Reference Image.
- ROI Rotation: the orientation of the ROI can be changed inside the Reference Image (only in the DataVS2 GUI 1.2 version).

The positions found are graphically represented inside the ROI with yellow arrows located in the found points. The searching direction is represented by a yellow arrow passing through the ROI barycentre. The searching direction can be of two types:

- from external ROI borders though the ROI barycentre; Mode: External.
- from ROI barycentre through external ROI borders; Mode: Internal.

If the difference between the positions found match the expected range, the ROI borders will be shown in green, representing a SUCCESS condition.
If no edges are found or if the calculated width is outside the expected limits, the ROI borders will be shown in red, representing a FAILURE condition.

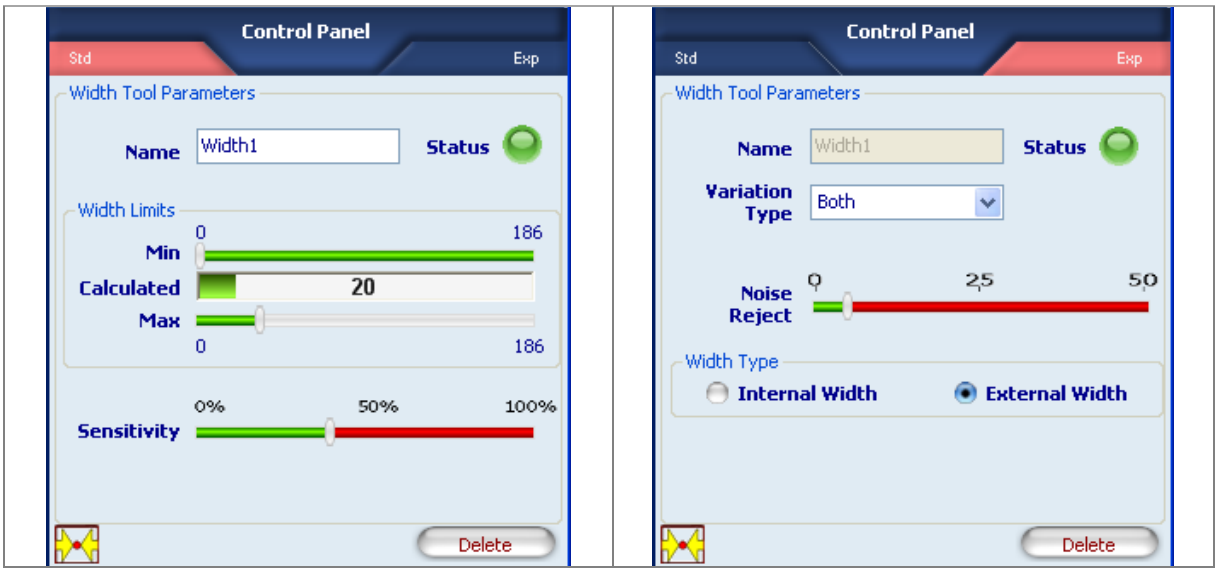
NOTE

The searching direction represented by the yellow line is always referred to the ROI top-left corner.

Width Control parameters

The following image and the table display the Position Locator parameters.

Standard Parameters	Expert Parameters

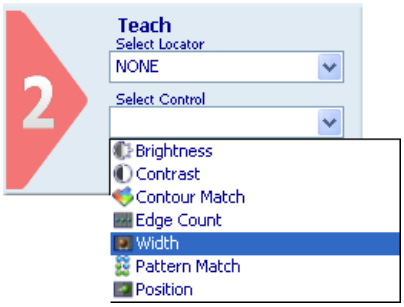


Parameters	Description	Type	Default
Name	The Name assigned to the Control Tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	Width
Status	A tool result indicator . • Green: SUCCESS. The measured width is within the range set in “Width limits”. • Red: FAILURE. Width hasn’t found any edge or the measured width doesn’t match “Width limits”	Standard, Output	
Calculated	Current Width measured by the Control inside the ROI along the searching direction.	Standard, Output	
Minimum	Minimum width inside the ROI, along the searching direction. This threshold must be compatible with the value found (shown near the "Calculated" label): this means Min must be less than or equal to "Calculated".	Standard, Input	0
Max	Maximum value of width inside the ROI, along the searching direction. This threshold must be compatible with the value found (shown near the "Calculated" label): this means Max must be more than or equal to "Calculated". ***Note***: as Width is calculated within ROI borders the Maximum width is always equal to ROI Width.	Standard, Input	ROI Width
Sensitivity	The degree of accuracy provided when searching for the Width. The lower the Sensitivity, the greater the accuracy.	Standard, Input	40%
Variation Type	Determines the type of edges to be searched inside the image. Four are the available options: • From white to black: only white to black transitions are detected; • From black to white: only black to white transitions are detected; • Automatic: both white to black and black to white transitions are detected; the returned one is the transition with the largest difference between the gray values. • Both: both white to black and black to white transitions are detected; the returned one is the first transition found.	Expert, Input	All

Parameters	Description	Type	Default
Noise Rejection	Reduces the effect that noisy pixels will have on the edge detection. The higher the value, the stronger the brightness variation needed to recognize an edge.	Expert, Input	5
Width type	Determines how the tool reaches for the edges that will represent the width limits. Two are the possible values: Internal: tool searches from the barycentre of the ROI to the ROI borders along the searching direction. External: tool searches from the ROI borders to the ROI barycentre along the searching direction.	Expert, input	External
Delete	Deletes the Locator from the current Inspection		

Example

- Add a "Width" edge to current inspection using the Graphic User Standard in STANDARD mode.
- Select "Width" from the pull-down menu:



- Position the control on the reference image in the work area: the pointer changes shape as soon as it enters in the work area. Left-click once on any position to create a default rectangular ROI.
- At the same time, the control features and the result of its application on the reference image appear in the control panel.
- Set ROI size and position as required, as explained above.
- The result indicated by STATUS is also highlighted by the ROI border changing colour.
- As soon as the values change, result is calculated without delay.
- Let us assume that our reference is the distance between points that are part of a bright object on a dark background.
- A yellow line appears inside the ROI on the edges found; edges are displayed as arrows that identify the limit values for distance calculation.
- Now use the sliders of the Min and Max fields to specify how far the measurements taken on master images are allowed to deviate from the measurement you have just set and still be considered acceptable (i.e. without triggering a control "Fail" status).

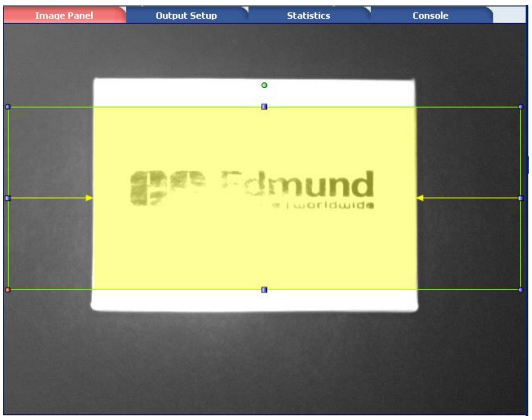


Fig. A: SUCCESS

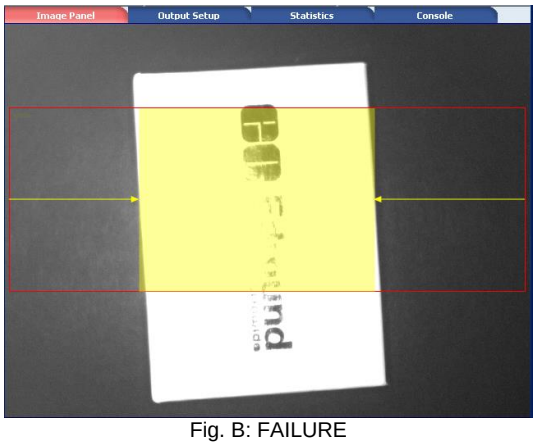


Fig. B: FAILURE

5.4.7 Pattern Match

In pattern match operations, target object and current object are compared to determine if they are similar.

The Pattern Match Control searches for all the occurrences of the target inside the search area and determines their position and the number of occurrences found. In this way it is possible to detect the horizontal translations and either the vertical displacements of the object in the inspection area.

The Pattern Match Control finds horizontal and vertical positions within the SEARCH AREA. It can be used to compensate rotational displacements up to a maximum of 5-10 degrees.

Possible uses are:

- Verifying the integrity of an object (i.e. a label on a box)
- Control of absence/presence of objects
- Counting of the number of objects equivalent to the target one and currently present inside the image

Pattern Match Control selection and positioning

You can select the Pattern Match Control from the Controls pull-down menu in STEP2. The Pattern Match Control is identified by the following icon:

Pattern Match Control is characterized by two different ROIs, having different scopes:

- SEARCH AREA: the outer rectangle, representing the portion of the image where the Control will search for the target object.
- TEMPLATE ROI: the inner rectangle, representing the target pattern to be searched in the region delimited by the search area. The TEMPLATE ROI cannot be positioned outside the SEARCH AREA.

When you are in STEP 2, the Pattern Match Control ROI allows the following variations:

- Move the SEARCH AREA and the TEMPLATE ROI: the position of the ROI can be changed in the Reference Image.
- SEARCH AREA and TEMPLATE ROI resizing: the width and the height of the ROI can be changed in the Reference Image
- SEARCH AREA rotating: the orientation of the Search Area can be changed in the Reference Image. Search Area rotation will affect also Template ROI rotation, when rotating the search area, you will see the template ROI synchronously rotating with it.

During the setup process, you need to define the target object to be identified. The system stores the brightness value of each composing pixel included in the TEMPLATE ROI.
During the inspection, the control will search for all the instances of the target in the SEARCH AREA. When it finds an instance of the target, it will highlight it with a green rectangle.
Degree of similarity and accuracy may be set from the control panel.

NOTE

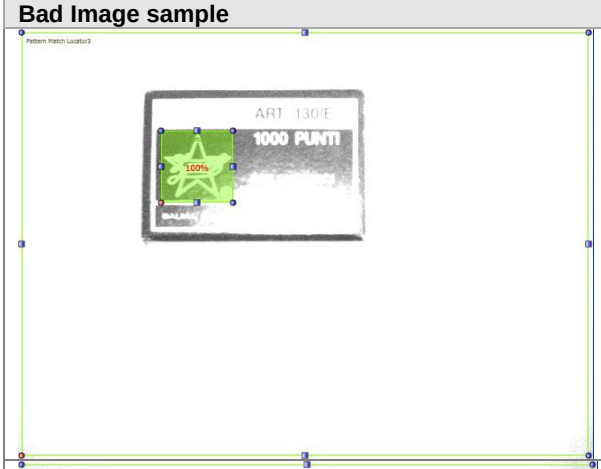
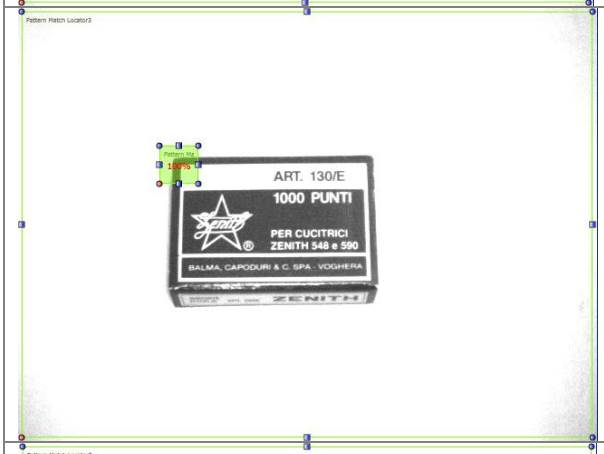
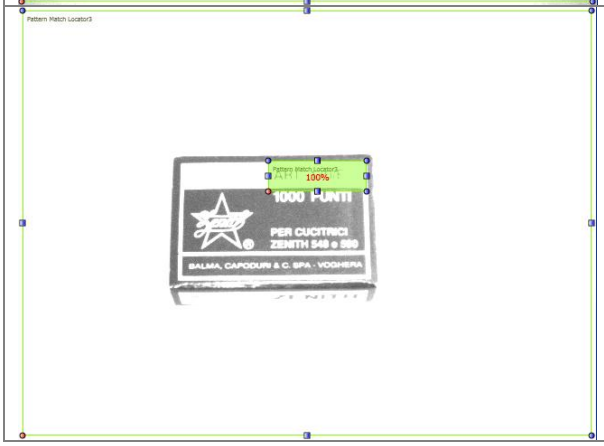
In order to have a reliable and robust behaviour of the Pattern Match Control we recommend you to configure the TEMPLATE ROI using a well contrasted reference image of the target object.

Some useful hints that can help you choose a good target object image:

- The reference image has to be in focus and have a good contrast and brightness level; you can adjust image brightness and contrast by changing the image acquisition Exposure Time and Gain in STEP1.
- The target object has to be completely included inside the template ROI borders; adjust TEMPLATE ROI dimensions and position in order to include only the object to be searched.
- The target object texture should be recognizable and in contrast with its background; assure the reference image doesn't contain ambiguous objects or characteristics having the same texture of the target object. Only the target object should be present in the reference image while the remaining part of the image (the background) should be uniform.

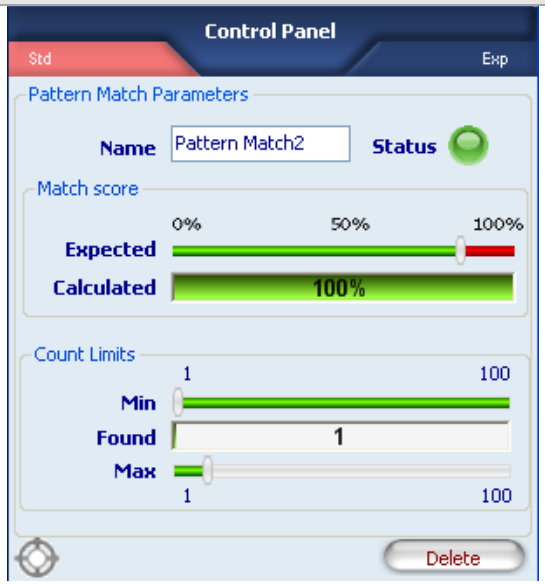
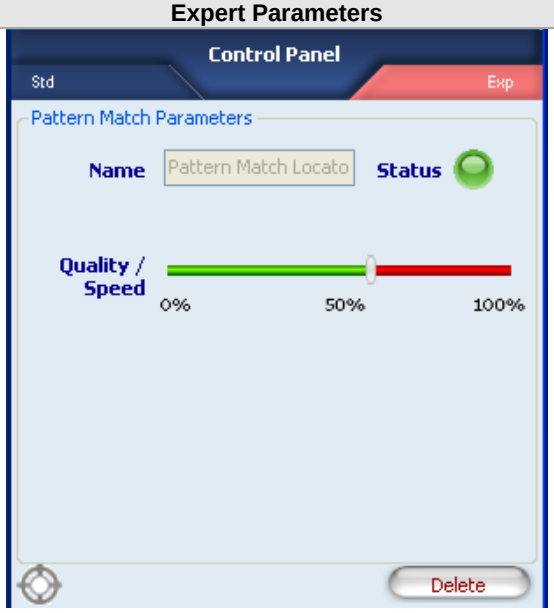
The DataVS2 GUI software helps you define a good quality TEMPLATE ROI: a popup message will be appear whenever the chosen target object is not suitable (see picture below).
When this message is shown, please try to adjust the TEMPLATE ROI by resizing or moving it. If nothing changes, we suggest you to choose a different reference image.

In the following table you will find some examples of a bad target object configuration:

Bad Image sample	Description
	The Reference Image is not in focus and hasn't a good contrast and brightness level. In this case you should change the Reference Image after adjusting focus and decreasing the Exposure Time.
	The target object (sensor Logo) is not completely included inside the TEMPLATE ROI borders. The dimensions of the template ROI should be adjusted in this case.
	The target object texture is not recognizable enough compared with its background. In this case you should change the TEMPLATE ROI after modifying the gain; this should increase the contrast in the image

Pattern Match Control parameters

The following image and the table display the Pattern Match Control parameters.

Standard Parameters	Expert Parameters
	

Parameters	Description	Type	Default
Name	The Name assigned to the Control Tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	Pattern Match
Status	A tool result indicator . • Green: SUCCESS. The Pattern Match Control has found at least one occurrence of the target pattern with a score higher that the expected one. • Red: FAILURE. Pattern Match Control hasn't found any occurrence of the target pattern or the score of the found pattern is lower than the expected one.	Standard, Output	
Calculated	The score representing the worst calculated degree of similarity of the current found objects to the reference target one.	Standard, Output	
Expected	Threshold used for the result evaluation. It indicates how closely the found pattern in the SEARCH AREA must resemble the target pattern in order to be considered valid. 100% means perfect match, 0% means no match found. ***Note***: Expected value change might involve a modification of found objects, and this depends on their calculated score.	Standard, Input	85%
Found	The number of found occurrences of the target object inside the image having a matching score higher than the expected one.	Standard, Output	
Accuracy / Speed	Influences the precision level used by the Pattern Match Tool during the searching phase; the higher the accuracy value, the higher will be the precision in the calculated score of the found object. Low level of Accuracy will speed up the searching process: we recommend to use low values of Accuracy when the execution time doesn't meet your requirements.	Expert, Input	60%
Delete	Deletes the Locator from the current Inspection		

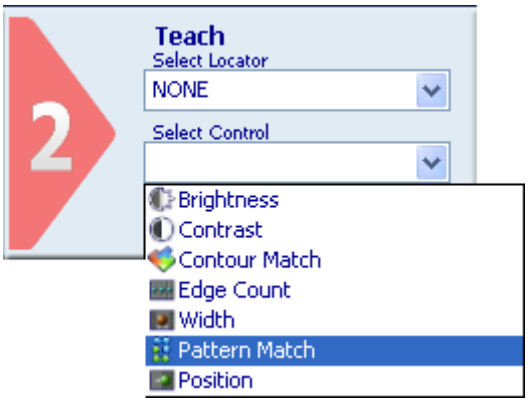
To quicken the Pattern Match Control execution time you can act on two different parameters:

- Reducing the SEARCH AREA dimensions: if you are sure the object to be found is always in a restricted area of the image, we recommend to reduce the SEARCH AREA dimension in order to cover only the portion of the image where you want the object to be searched. This will reduce time needed to locate the object (as the portion of the image where to search for the target pattern is smaller) and will decrease the risk of errors due to noise inside the image.
- Reducing the Accuracy value: reducing the level of precision required to the Pattern Match Tool during the searching phase will decrease the time needed to calculate the match score.

NOTE *When the number of found occurrences of the target object doesn't find the expected limits the tool will show FAIL result and only a Red ROI will be shown. You can check the calculated values (worst match found and the number of occurrences found) in the parameter panel.*

Example

- Add a "Pattern Match" control to current inspection.
- Select "Pattern Match" from the pull-down menu:



- Position the control on the reference image in the work area: the pointer changes shape as soon as it enters in the work area. Left-click once on any position to create a default rectangular ROI.
- At the same time, the control features and the result of its application to the reference image appear in the control panel.
- Set ROI size and position as required, as explained above.
- The result indicated by STATUS is also highlighted by the ROI border changing colour.
- The value of parameter "Calculated", which indicates the degree of similarity in percentage, is reported inside the areas inside the search ROI found to be similar to the master image of the reference ROI.
- As soon as the values change, result is calculated without delay.
- Another possible application example of the Pattern Match Control Tool is shown below.
- The tool is used to count the number of bottles present inside a box. Fig. A shows a successful check, on the contrary Fig. B shows a failure: a hand obstructs the view and the Control fails.

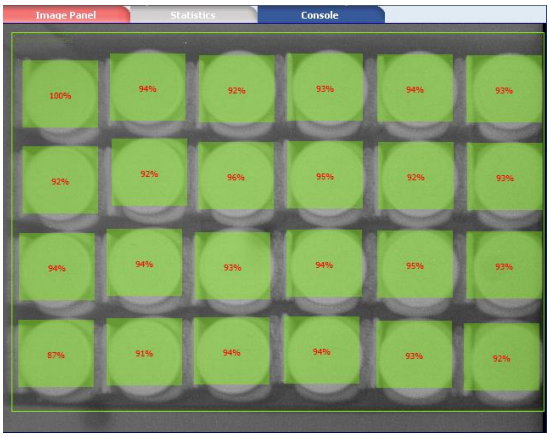


Fig. A: SUCCESS

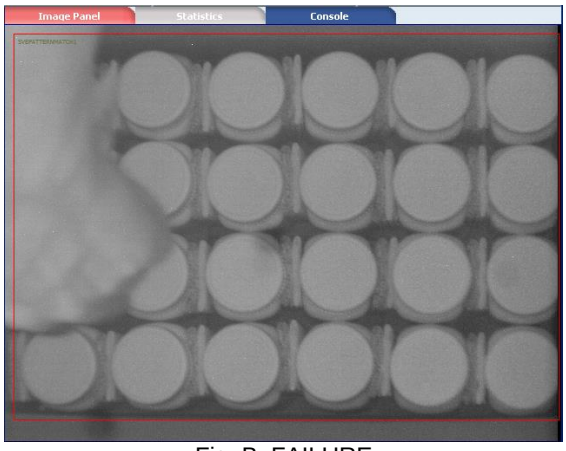


Fig. B: FAILURE

5.4.8 Position

Position Control can be used to check the position of an object inside a ROI.
This control works searching for a gray scale level variation (edge detection ►) along a well-defined direction inside the ROI. An Edge is a defined border between a light area and a dark area in the image.
The Position Control identifies the transition between:

- A light and a dark area (White to Black)
- A dark and a light area (Black to White)

in the current image.
Based on the first derivative of the pixel brightness variation function, the maximum positive peak is identified if control is targeting a positive edge, or the maximum negative peak is identified when targeting a negative edge. A point is considered an edge point when its brightness variation exceeds this threshold, which is a function of the peak value found;
For instance, imagine a dark object on a light background: object contour is identified as the edge, i.e. the set of dark pixels lying next to light pixels. However, it takes a sharp tone variation for the light-to-dark transition to be interpreted as an edge; the extent of the transition is determined by a threshold (Edge detection ►).
Possible uses are:

- Verifying the integrity of an object (i.e. the correct placement of a beak on a bottle)
- Control of absence/presence of objects

Position Control selection and positioning

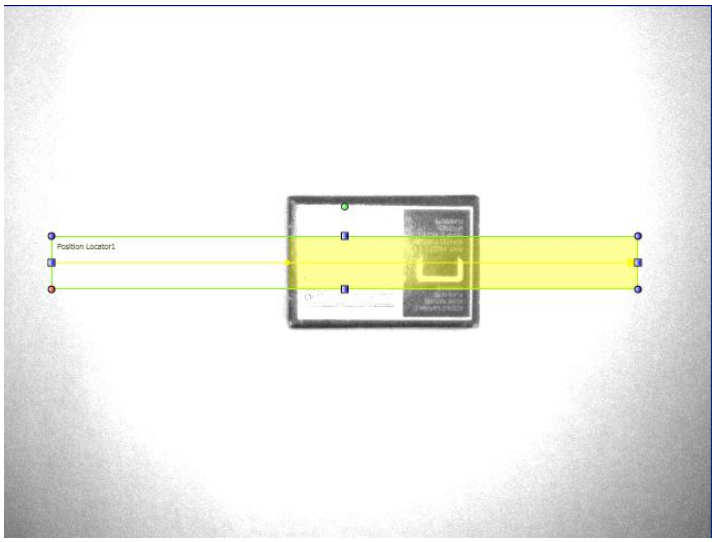
You can select the Position Control from the Controls pull-down menu in STEP2. The Position Control is identified by the following icon:

When you are in STEP 2 the Position Control ROI allows the following variations:

- ROI Moving: the position of the ROI can be changed in the Reference Image.
- ROI resizing: the width and the height of the ROI can be changed in the Reference Image.
- ROI Rotation: the orientation of the ROI can be changed in the Reference Image.

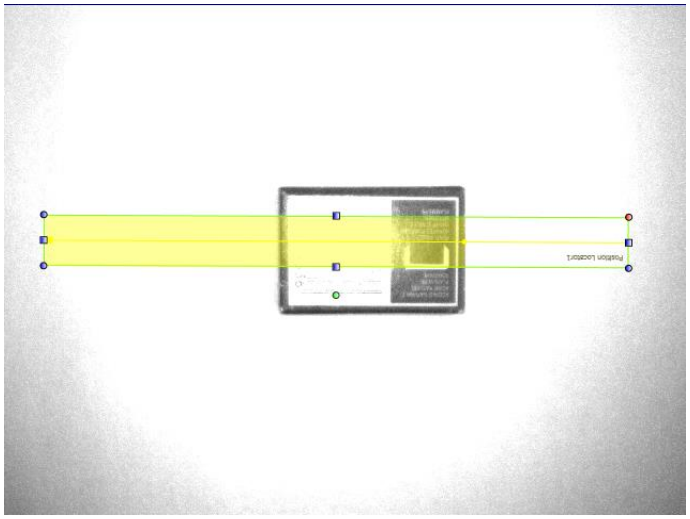
The found edge is graphically represented inside the ROI with a yellow dot located in the matching point. The searching direction and versus is represented by a yellow arrow passing through the ROI barycentre and going from Left to Right.
If the edge position matches the “Position limits” range, the ROI borders will be shown in green, representing a SUCCESS condition.
If no edges are found or if the found position is outside “Position limits”, the ROI borders will be in red, representing a FAILURE condition.

After the Variation Type of the edge (White to Black or Black to White) to be identified has been specified, the Locator processes the ROI in the direction shown by the yellow arrow and stops as soon as it comes across an edge that meets the set specifications.



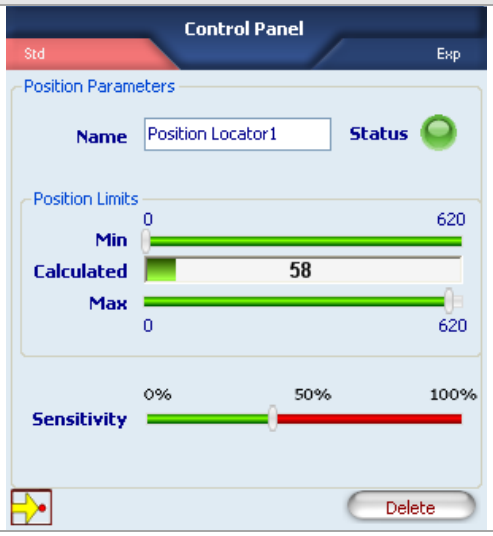
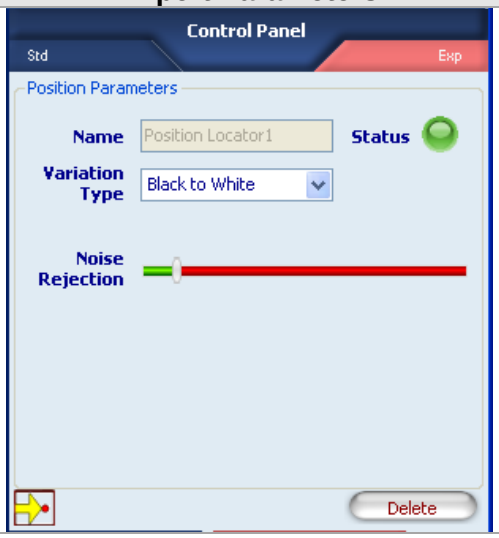
NOTE

The searching direction represented by the yellow line is always referred to the ROI top-left corner. When the ROI orientation is 180 degree you will see the yellow line going from Right to Left.



Position Control parameters

The following image and the table display the Position Control parameters.

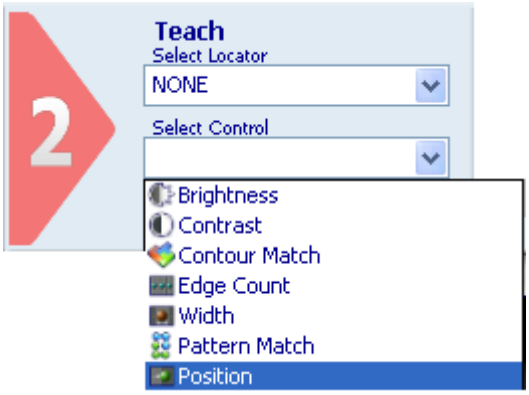
Standard Parameters	Expert Parameters
	

Parameters	Description	Type	Default
Name	The Name assigned to the Control tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	Position
Status	Tool that indicates a result: • Green: SUCCESS. The Position Control has detected an Edge within the range set in “Position limits”. • Red: FAILURE: Position Control has not found any edge or the found Edge does not match “Position limits”	Standard, Output	
Calculated	Current position of the edge found by the position Control inside the ROI along the searching direction.	Standard, Output	
Minimum	Minimum value of detectable position inside the ROI, along the searching direction. This threshold must be compatible with the value found (shown near the "Calculated" label): this means Min must be less than or equal to "Calculated".	Standard, Input	0
Max	Maximum value of detectable position inside the ROI, along the searching direction. This threshold must be compatible with the value found (shown near the "Calculated" label): this means Max must be more than or equal to "Calculated". ***Note***: Min and Max always refer to the ROI area. 0 (Min) is always considered the left border of the ROI, the value shown as Max is always considered the right border of the ROI and represent the ROI width in pixels. Looking to the picture above ROI width is 620 pixels.	Standard, Input	Dimensions ROI
Sensitivity	The degree of accuracy provided when searching for the Edge. The lower the Sensitivity, the greater the accuracy.	Standard, Input	40%
Variation Type	Determines the type of edges to be searched inside the image. Four are the available options: • White to Black: only white to black transitions are detected; • Black to White: only black to white transitions are detected; • Auto: both white to black and black to white transitions are detected; the returned one is the transition with the largest difference between the gray values. • Both: both white to black and black to white transitions are detected; the returned one is the first transition found.	Expert, Input	All

Parameters	Description	Type	Default
Noise Rejection	Reduces the effect that noisy pixels will have on the edge detection. The higher the value is, the stronger the brightness variation needed to recognize an edge.	Expert, Input	5
Delete	Deletes the Control from the current Inspection.		

Example

- Add a "Position" control to current inspection to identify two edges.
- Select "Position" from the pull-down menu:



- Position the control on the reference image in the IMAGE PANEL: the pointer changes shape as soon as it enters in the work area. Left-click once on any position to create a default rectangular ROI.
- At the same time, the control characteristics and the result of its application on the reference image appear in the control panel.
- Set ROI size and position as required, as explained above.
- The result indicated by STATUS is also highlighted by the ROI border changing colour.
- As soon as the values change, result is calculated without delay.
- Set up system to recognize all dark-to-light brightness variations as edges. A yellow line appears inside the ROI on the edges found; edges are displayed as yellow dots that identify the limit values for distance calculation.
- Now use the sliders of the Min and Max fields to specify how far the measurements taken on master images are allowed to deviate from the measurement you have just set and still be considered acceptable (i.e. without triggering a control "Fail" status).
- In this example a possible usage scenario is presented. In this particular task, it's necessary to verify the correct placement of the beak of a bottle of water-closet detergent. As shown in Fig. A and B, the Position Control Tool is able to carry out this operation.
- The Fig. A shows a correct placement, on the contrary the Fig. B shows a wrong placement.

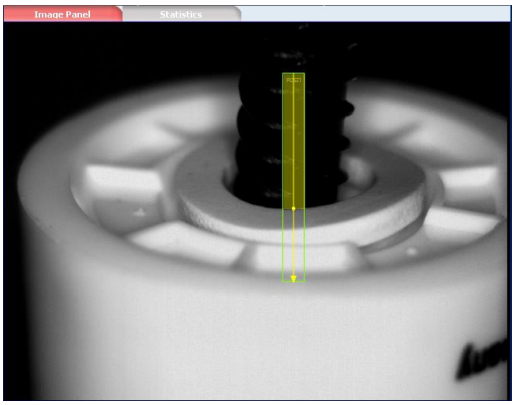


Fig. A: SUCCESS

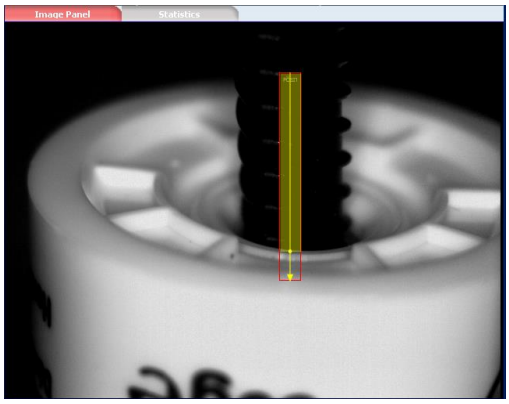


Fig. B: FAILURE

5.4.9 Datamatrix

The Datamatrix control searches and decodes the contents of one or more Datamatrix symbols inside a Region of Interest.
The Control can be used in the following applications:

- Reading of contents of one or more Datamatrix symbols inside a ROI;
- Checking of information contents of one or more Datamatrix symbols inside a ROI;
- Counting of the number of Datamatrix symbols inside a ROI.

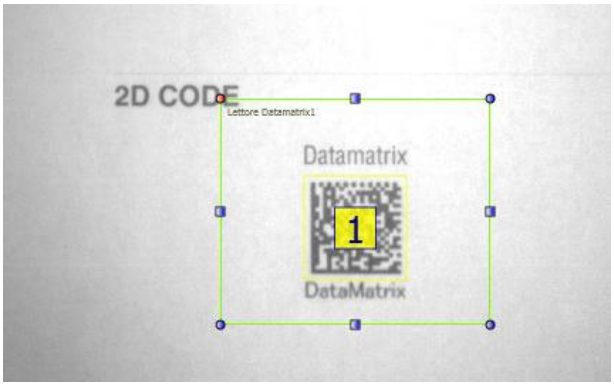
Datamatrix control selection and positioning

You can select the Datamatrix Control from the drop-down Control menu in STEP2. The Datamatrix Control is identified by the following icon:

When you are in STEP 2 the Datamatrix Control allows you to make the following changes:

- ROI Moving: the position of the ROI can be changed in the Reference Image.
- ROI resizing: the width and the height of the ROI can be changed in the Reference Image.

The symbol found is graphically represented inside the ROI with a yellow frame that identifies the vertexes and a rectangle containing the index of the code found, visualized in the centre of code itself. The index, in a decimal format, is displayed to distinguish the Datamatrix symbols found inside the same ROI.



The SUCCESS or FAILURE control result depends on the functioning mode (Control Type) selected. Three functioning modes are accepted by this control:

- **Decoding**: in this mode the Datamatrix Control decodes all the Datamatrix symbols in the ROI. The contents of the symbols decoded is displayed in the **Read Code** area. This Control provides, for each symbol decoded, the contents as well as the corresponding Quality level of each code read, represented by a set of parameters displayed in the **Quality** area. The result sent is SUCCESS if at least one symbol is decoded in the ROI and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure.
- **String Matching**: in this mode the Datamatrix Control compares the contents of the decoded symbols with one or more character strings listed in the **Expected Code** field. The result sent is SUCCESS if at least one occurrence of each **Expected Code** is found and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure.
- **Counting**: in this mode the Datamatrix Control verifies that the number of decoded symbols in the ROI belongs to the interval specified in the **Counting Limits** field. The result sent is SUCCESS if the number of symbols decoded is between the Minimum and Maximum values specified and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure.

Datamatrix Symbol Characteristics

The Datamatrix control of DataVS2 decodes the Datamatrix symbols with the following characteristics:

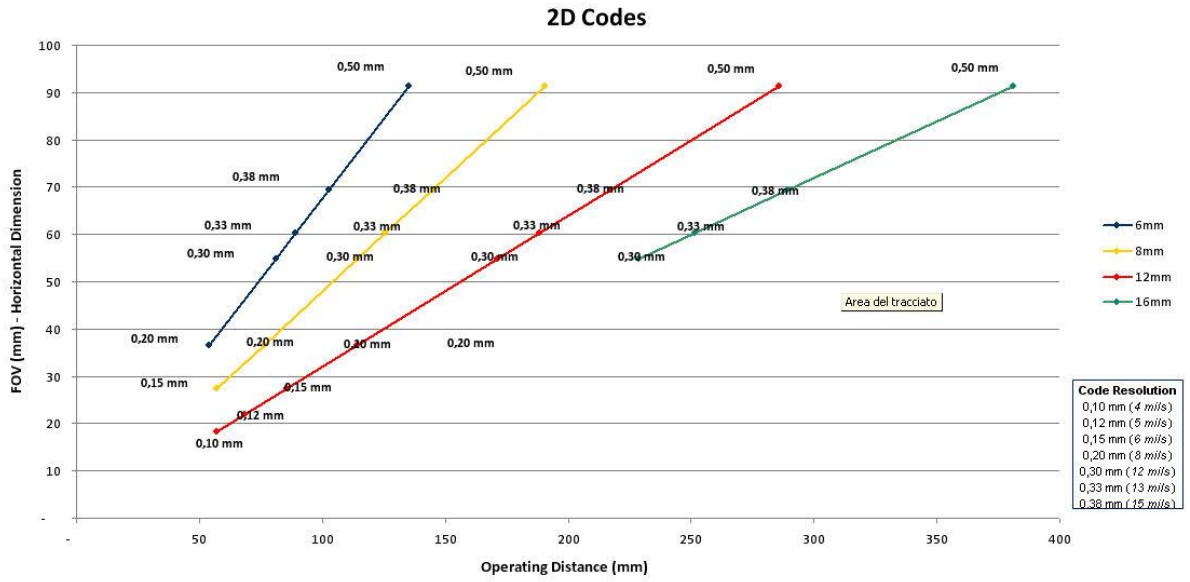
- **Size**: the number of rows and columns composing a Datamatrix symbol that can be decoded using DataVS2 can vary from 8x8 to 144x144. All possible value ranges are accepted for a Datamatrix symbol.
- **Coding**: no limits are foreseen relative to the type of coding used for the data inside the symbol. In the result string sent by DataVS2 on the Ethernet and Serial channels, the code read is represented in a byte vector format; the byte decoding is made by the client that reads it.

NOTE

DataVS2 GUI decodes all bytes of the code read according to the ASCII table. The contents of a Datamatrix symbol that has not been coded in an ASCII format cannot be displayed.

- **Resolution**: the number of pixels per symbol module to decode can vary in the [3, 20] range. This value represents the resolution that characterizes the symbol to decode inside an image.

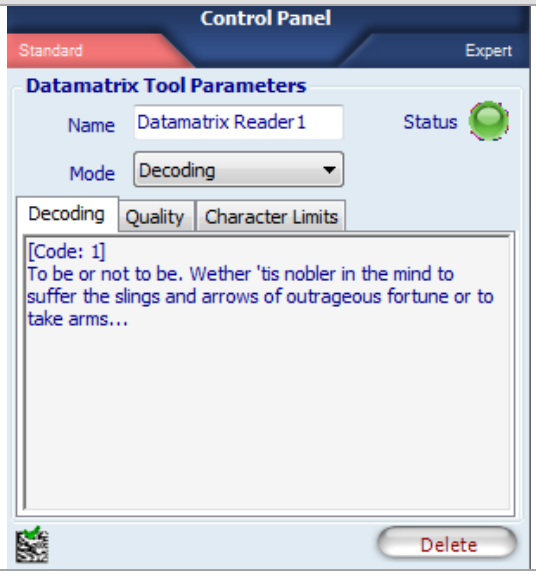
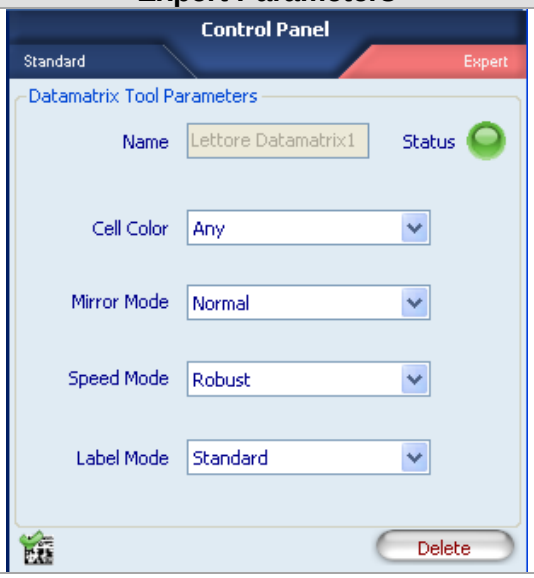
Please refer to the following graphic for more information on the resolution of the code to read, expressed in mm/FOV:



- **Orientation Angle**: the symbol can have all the possible orientation angles in the ROI.
- **Edges (Quiet Zone):** Datamatrix symbols having Edge dimensions equal to 1 module (the dimension of a unit containing information inside the symbol) are accepted.

Datamatrix Control Parameters

The following tables show the Datamatrix Control parameters:

Standard Parameters	Expert Parameters
	

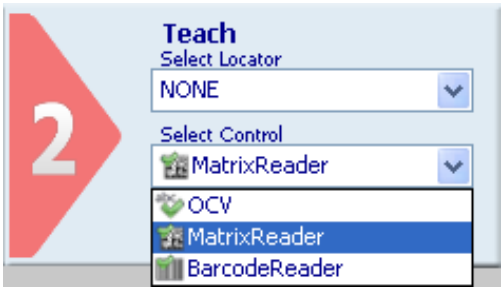
Parameters	Description	Type	Default
Name	The Name assigned to the Control tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	Datamatrix
Status	Tool that indicates a result: • Green: SUCCESS. The Datamatrix Control has decoded one or more symbols in the ROI and these correspond to the control filters (Control Type) set. • Red: FAILURE. Datamatrix Control hasn't decoded any symbols in the ROI or the decoded symbol is incompatible with the filters set.	Standard, Output	
Mode	The **Mode** parameter selects the functioning mode of the Datamatrix Control. Three functioning modes can be selected: • **Decoded**: in this mode the Datamatrix Control decodes all the Datamatrix symbols in the ROI. The contents of the symbols decoded is displayed in the **Read Code** area. This Control provides, for each symbol decoded, the contents as well as the corresponding Quality level of each code read, represented by a set of parameters displayed in the **Quality** area. The result sent is SUCCESS if at least one symbol is decoded in the ROI and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure. • **String Matching**: in this mode the Datamatrix Control compares the contents of the decoded symbols with one or more character strings listed in the **Expected Code** field. The result sent is SUCCESS if at least one occurrence of each **Expected Code** is found and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure. • **Counting**: in this mode the Datamatrix Control verifies that the number of decoded symbols in the ROI belongs to the interval specified in the **Counting Limits** field. The result sent is SUCCESS if the number of symbols decoded is between the Minimum and Maximum values specified and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure.	Standard, Input	Decoded
Decoding	The **Decoding** field displays the contents of each symbol decoded. Each contents is identified by the **[Code i]** prefix where **i** represents the corresponding code index displayed in the Image Panel.	Standard, Output	
Quality	These values represent the general quality of the decoded symbol measured by the Datamatrix Control. The quality of a Datamatrix symbol is represented by the measurement of the following parameters, expressed according to the [0, 1, 2, 3, 4] value scale, where 4 corresponds to the maximum quality value of a code: • **Symbol Contrast**: represents the symbol contrast level expressed as the difference of reflectivity between light and dark parts of the symbol. • **Axial Non Uniformity**: represents the average space difference between two sides of the symbol arranged along the X and Y axes. • **Unused Error Correction**: is the redundancy measurement used for the correction of errors inside the same symbol. • **Decode Grade**: represents the result supplied by the	Standard, Output	

Parameters	Description	Type	Default
	reference decoding algorithm used by the Datamatrix Control. • **Overall Grade**: is equal to the minimum conformity value detected in the quality parameters described above. This value represents the general quality of the symbol and is also graphically displayed next to the **Calculated** label. • The **Expected Value** parameter of the Quality field can be used to set a filter on the expected Quality value. The Datamatrix control, in each execution, verifies that the **Calculated Value** is higher or equal to the **Expected Value**.		
Character Limits	It is possible to insert a filter on the number of read characters. The codes which have a number of characters greater than the maximum or less than the minimum are not returned by the tool.	Standard, Input	[0, 3116]
Expected Code	Allows to specify the expected information contents of one or more symbols to decode. To insert the Expected Code, the user just has to enter the corresponding string of characters in the dedicated area of the control panel. Wild card characters can be used to specify any sequence of characters within the expected string: • "?": use this symbol to specify any character in a known position of the expected string; any decoded symbol corresponding to the position of the jolly symbol will be considered correct. • "*": use this symbol to specify any sequence of characters in a known position of the expected string: any decoded symbol sequence corresponding to the position of the jolly symbol will be considered correct. More than one Expected Code can be specified for each Datamatrix Control; in this case the Control verifies that each decoded code content matches at least one expected code string. The Teach-in of the Expected Code can be set through the **Auto** selection box; the Control will set the decoded symbol code present in the Reference Image as the Expected Code.	Standard, Input	-
Counting limits	Allows to specify the minimum and maximum number of Datamatrix symbols that have to be present in the ROI for the Control result to be SUCCESS. These parameters are enabled only when the **Counting** mode is selected: • **Calculated**: represents the number of Datamatrix symbols decoded in the ROI. • **Minimum**: represents the minimum number of Datamatrix symbols that have to be decoded in the ROI for the Control result to be SUCCESS. • **Maximum**: represents the maximum number of Datamatrix symbols that have to be decoded in the ROI for the Control result to be SUCCESS.	Standard, Input	Minimum: 0, Maximum: 100.
Cell Colour	Represents the colour of the Datamatrix symbol to decode: • **Black on White**: black on white background. • **White on Black**: white on black background. • **Any**: the Control in this case tries to decode the symbol according to both the previous options.	Expert, Input	Any
Mirror Mode	Defines the orientation of the cells respect to the **Finder Pattern** of the symbol. The codes can be Normal or Mirrored.	Expert, Input	Normal
Speed Mode	Influences the accuracy level used by the Datamatrix during the search phase; the higher the accuracy value, the more precise the calculation of the similarity degree of the object found. Low levels of Accuracy increase the searching process speed: we suggest to use low Accuracy values when the execution time does not respect the requirements.	Expert, Input	Robust
Label Mode	Defines the print typology of the symbol, that has to be	Expert, Input	Standard

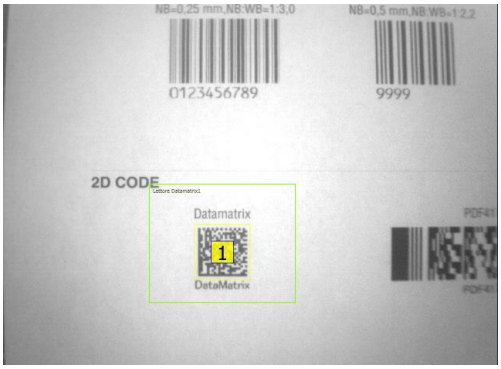
Parameters	Description	Type	Default
	found among the following alternatives: • **Standard**: inkjet printing; in this case the symbol present high contrast. • **Dotpeen**: direct marking print. • **Low Contrast**: inkjet or direct marking print characterized by a low contrast between symbol colour and background.		
Delete	Deletes the Control from the current Inspection.		

Example

- Create an Inspection to verify the contents of a Datamatrix symbol.
- Assure that the DataVS2 ID is selected for the current Inspection and select “Datamatrix” from the drop-down menu of the available Controls.



- Position the control on the reference image in the IMAGE PANEL: the pointer changes shape as soon as it enters in the work area. Left-click once on any position to create a default rectangular ROI.
- At the same time, the control characteristics and the result of its application on the reference image appear in the control panel.
- Assure that the reference image contains the Datamatrix symbol to decode and that it corresponds to the characteristics specified above.
- Set ROI size and position as required. The symbol to be decoded has to be always inside the rectangle visualized.
- The result indicated by STATUS is also highlighted by the ROI border changing colour; the Datamatrix in the first positioning is configured for a Decoding type control; the result will be SUCCESS if the symbol is decoded If the Control FAILS, check that the ROI has been correctly positioned and that the symbol is clearly visible inside the image, possibly choosing a better reference image of the part to inspect.
- Now change the Control Type parameter selecting the String Control value; then select the Auto box: after this phase, the Control sets the contents of the symbol present in the reference image as expected value. From now onwards, the control will always verify that, from time to time, the contents of the coded symbol are equal to the expected ones.
- It is then possible to test the Inspection configured with images containing different symbols.



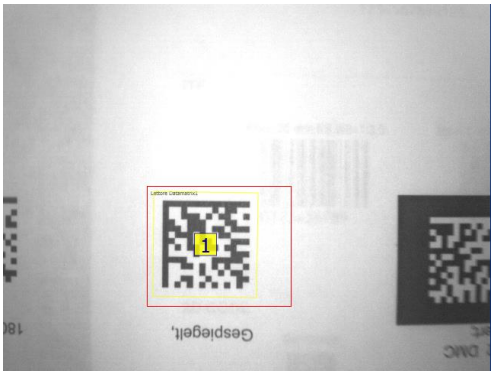


Fig. A: SUCCESS Fig. B: FAILURE

5.4.10 Barcode

The Barcode control searches and decodes the contents of one or more Barcode symbols inside a Region of Interest.
The Control can be used in the following applications:

- Reading of the contents of one or more Barcode symbols inside a ROI;
- Verification of the information contents of one or more Barcode symbols inside a ROI;
- Counting of the number of Barcode symbols inside a ROI.

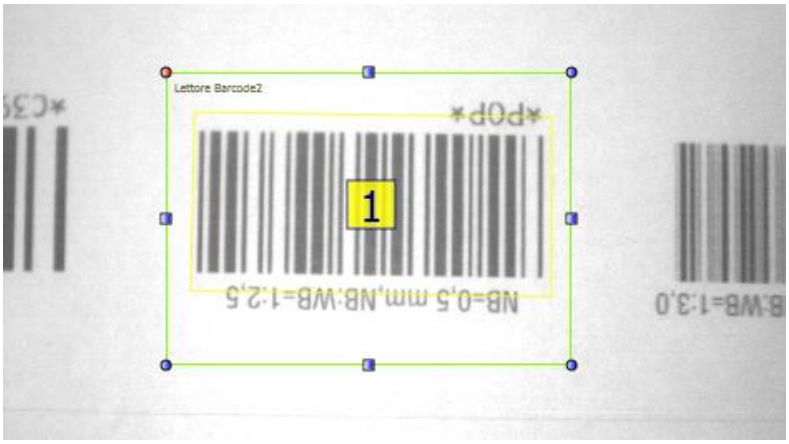
Barcode Control selection and positioning

You can select the Barcode Control from the drop-down Control menu in STEP2. The Barcode Control is identified by the following icon:

When you are in STEP 2 the Barcode Control allows you to make the following changes:

- ROI moving: the position of the ROI can be changed in the Reference Image.
- ROI resizing: the width and the height of the ROI can be changed in the Reference Image.
- ROI Rotation: the orientation angle of the ROI can be changed in the Reference Image.

The symbol found is graphically represented inside the ROI with a yellow frame that identifies the vertexes and a rectangle containing the index of the code found, visualized in the centre of code itself. The index, in a decimal format, is displayed to distinguish the Barcodes symbols found inside the same ROI.



The SUCCESS or FAILURE control result depends on the functioning mode (Control Type) selected. Three functioning modes are accepted by this control:

- **Decoding**: in this mode the Barcode Control decodes all the Barcode symbols in the ROI.
  The contents of the symbols decoded is displayed in the **Read Code** area. This Control provides, for each symbol decoded, the contents as well as the corresponding Quality level of each code read, represented by a set of parameters displayed in the **Quality** area. The result sent is SUCCESS if at least one symbol is decoded in the

ROI and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure.

- **String Matching**: in this mode the Barcode Control compares the contents of the decoded symbols with one or more character strings listed in the **Expected Code** field. The result sent is SUCCESS if at least one occurrence of each **Expected Code** is found and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure.
- **Counting**: in this mode the Barcode Control verifies that the number of decoded symbols in the ROI belongs to the interval specified in the **Counting Limits** field. The result sent is SUCCESS if the number of symbols decoded is between the Minimum and Maximum values specified and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure.

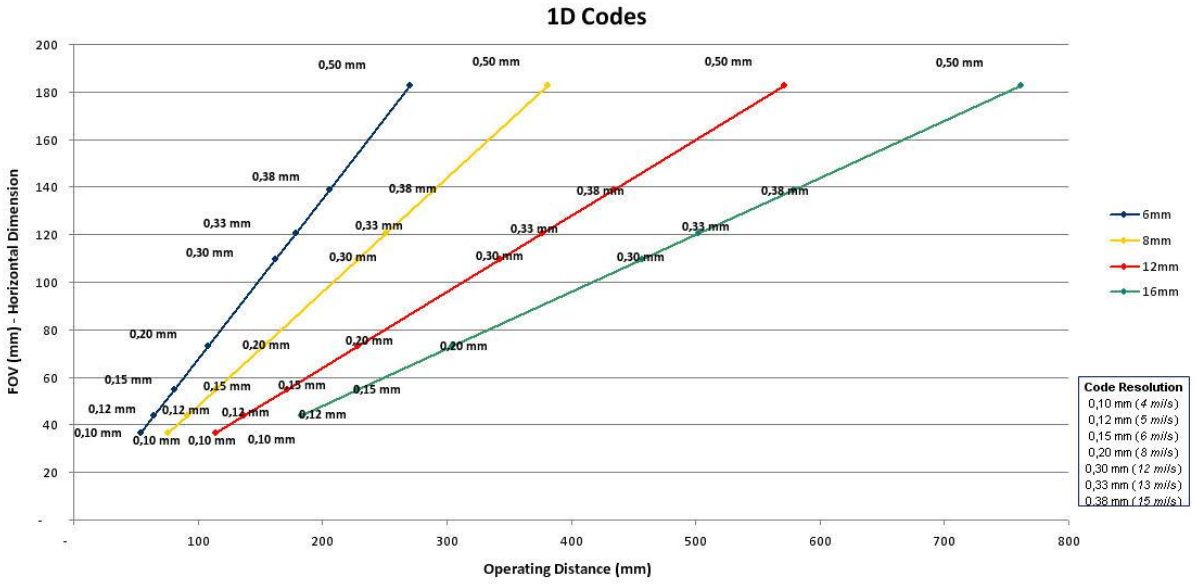
Barcode Symbol Characteristics

The Barcode control of DataVS2 decodes the Barcode symbols with the following characteristics:

- **Coding**: no limits are foreseen relative to the type of coding used for the data inside the symbol. In the result string sent by DataVS2 on the Ethernet and Serial channels, the code read is represented in a byte vector format; the byte decoding is made by the client that reads it.

NOTE *DataVS2 GUI decodes all bytes of the code read according to the ASCII table. The contents of a Barcode symbol that has not been coded in an ASCII format cannot be displayed.*

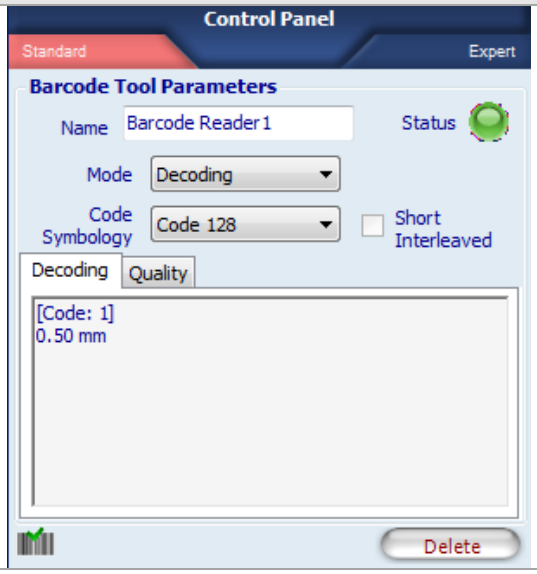
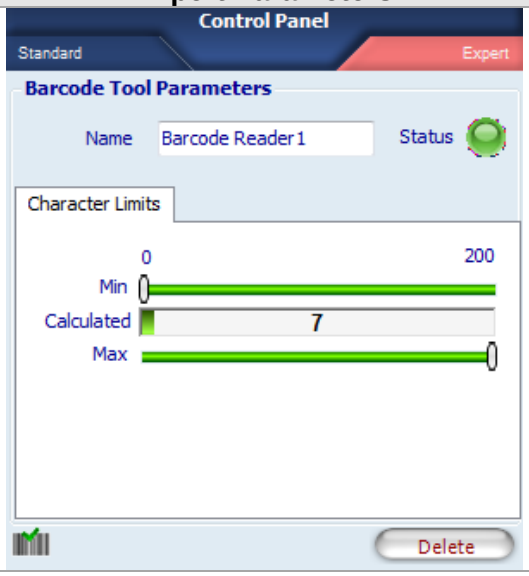
- **Resolution**: the following graphic provides more information on the resolution of the code to read, expressed in mm/FOV:



- **Orientation Angle**: the symbol can have all the possible orientation angles in the ROI.
- **Edges (Quiet Zone):** Barcode symbols having Edge dimensions equal to 1 module (the dimension of a unit containing information inside the symbol) are accepted.

Barcode Control Parameters

The following table shows the Barcode Control parameters:

Standard Parameters	Expert Parameters
	

Parameters	Description	Type	Default
Name	The Name assigned to the Control tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	Barcode
Status	Tool that indicates a result: • Green: SUCCESS. The Barcode Control has decoded one or more symbols in the ROI and these correspond to the control filters (Control Type) set. • Red: FAILURE. Barcode Control hasn't decoded any symbols in the ROI or the decoded symbol is incompatible with the filters set.	Standard, Output	
Mode	The **Mode** parameter selects the functioning mode of the Barcode Control. Three functioning modes can be selected: • **Decoding**: in this mode the Barcode Control decodes all the Barcode symbols in the ROI. The contents of the symbols decoded is displayed in the **Read Code** area below. This Control provides, for each symbol decoded, the contents as well as the corresponding Quality level of each code read, represented by a set of parameters displayed in the **Quality** area. The result sent is SUCCESS if at least one symbol is decoded in the ROI and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure. • **String Matching**: in this mode the Barcode Control compares the contents of the decoded symbols with one or more character strings listed in the **Expected Code** field. The result sent is SUCCESS if at least one occurrence of each **Expected Code** is found and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure. • **Counter**: in this mode the Barcode Control verifies that the number of decoded symbols in the ROI belongs to the interva specified in the **Counting Limits** field. The result sent is	Standard, Input	Decoded

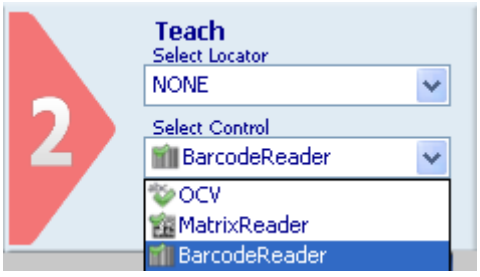
Parameters	Description	Type	Default
	SUCCESS if the number of symbols decoded is between the Minimum and Maximum values specified and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure.		
Decoding	The **Decoding** field displays the contents of each symbol decoded. Each contents is identified by the **[Code i]** prefix where **i** represents the corresponding code index displayed in the Image Panel.	Standard, Output	
Code Symbology	The Code Symbology field defines the type of Barcode code searched in the ROI. When no value is selected, the Control automatically sets the Symbology retrieving it from the code read in the Teach phase (Step 2). The possible symbologies accepted for a Barcode that can be decoded with DataVS2 ID or PRO are given below: • Code 128 • Code 39 • Codabar • Interleaved • EAN 13 • EAN 8 • UPC E • Postnet • IMB • Pharmacode • PDF 417 • UPC A • GS1-128 • Code 93 Note: for Pharmacode codes is necessary to manually set the parameter Symbology.	Standard, Input	Auto
Short Interleaved	It is possible to enable this parameter only if it's selected the Interleaved symbology. If enabled, you can read codes with only two characters, otherwise the minimum is five characters.	Standard, Input	Disabled
Expected Code	Allows to specify the expected information contents of one or more symbols to decode. To insert the Expected Code, the user just has to enter the corresponding string of characters in the dedicated area of the control panel. Wild card characters can be used to specify any sequence of characters within the expected string: • "?": use this symbol to specify any character in a known position of the expected string; any decoded symbol corresponding to the position of the jolly symbol will be considered correct. • "*": use this symbol to specify any sequence of characters in a known position of the expected string: any decoded symbol sequence corresponding to the position of the jolly symbol will be considered correct. More than one Expected Code can be specified for each Barcode Control; in this case the Control verifies that each decoded code content matches at least one expected code string. The Teach-in of the Expected Code can be set through the **Auto** selection box; the Control will set the decoded symbol code present in the Reference Image as the Expected Code.	Standard, Input	
Counting limits	Allows to specify the minimum and maximum number of Barcode symbols that have to be present in the ROI for the Control result to be SUCCESS. These parameters are enabled only when the **Counting** mode is selected: • **Calculated**: represents the number of Barcode symbols decoded in the ROI. • **Minimum**: represents the minimum number of Barcode symbols that have to be decoded in the ROI for the Control result to be SUCCESS.	Standard, Input	Minimum: 0, Maximum: 100.

Parameters	Description	Type	Default
	• **Maximum**: represents the maximum number of Barcode symbols that have to be decoded in the ROI for the Control result to be SUCCESS.		
Quality	These values represent the general quality of the decoded symbol measured by the Barcode Control. The quality of a Barcode symbol is represented by the measurement of the following parameters, expressed according to the [0, 1, 2, 3, 4] value scale, where 4 corresponds to the maximum quality value of a code: (*) • Symbol Contrast: the symbol contrast is the ratio between the lowest reflectivity of a bar and the highest reflectivity of a space. The higher the contrast, the best the symbol quality. • Minimum Edge Contrast: represents the minimum reflectivity. The lower this value (dark bars), the better the contrast. • Modulation: modulation is the ratio between Minimum Edge Contrast and Symbol Contrast. • Decodability: represents the result sent by the reference decoding algorithm used by the Barcode Control. The decoding is a prerequisite for the symbol quality check. • Overall Grade: is equal to the minimum conformity value detected in the quality parameters described above. This value represents the general quality of the symbol and is also graphically displayed next to the **Calculated** label. • The **Expected Value** parameter of the Quality field can be used to set a filter on the expected Quality value. The Barcode control, in each execution, verifies that the **Calculated Value** is higher or equal to the **Expected Value**.	Expert, Output	
Character number limits	Allows to specify the minimum and maximum number of characters that have to be present in the symbol for the Control result to be SUCCESS. • **Calculated**: represents the number of characters decoded in the symbol. • **Minimum**: represents the minimum number of characters that have to be decoded in the symbol for the Control result to be SUCCESS. • **Maximum**: represents the maximum number of characters that have to be decoded in the symbol for the Control result to be SUCCESS.	Expert, Input	Minimum: 0, Maximum: 200.
Delete	Deletes the Control from the current Inspection.		

(*) Exceptions are the following symbologies: IMB, Postnet, PDF417 and Pharmacode. For these symbologies all parameters are not calculated.

Example

- Create an Inspection to verify the contents of a Barcode symbol.
- Assure that the DataVS2 ID is selected for the current Inspection and select “Barcode Reader” from the drop-down menu of the available Controls.



- Position the control on the reference image in the IMAGE PANEL: the pointer changes shape as soon as it enters in the work area. Left-click once on any position to create a default rectangular ROI.
- At the same time, the control characteristics and the result of its application on the reference image appear in the control panel.
- Assure that the reference image contains the Barcode symbol to decode and that it corresponds to the characteristics specified above.
- Set ROI size and position as required. The symbol to be decoded has to be always inside the rectangle visualized.
- The result indicated by STATUS is also highlighted by the ROI border changing colour; the Barcode in the first positioning is configured for a Decoding type control; the result will be SUCCESS if the symbol is decoded. If the Control FAILS, check that the ROI has been correctly positioned and that the symbol is clearly visible inside the image, possibly choosing a better reference image of the part to inspect.
- Now change the Control Type parameter selecting the String Control value; then select the Auto box: after this phase, the Control sets the contents of the symbol present in the reference image as expected value. From now onwards, the control will always verify that, from time to time, the contents of the coded symbol are equal to the expected ones.
- It is then possible to test the Inspection configured with images containing different symbols.

Fig. A: SUCCESS

Fig. B: FAILURE

NOTE

It is possible to add to an inspection a maximum of 15 Barcode controls.

5.4.11 QRCode

The QRCode control searches and decodes the contents of one or more QR symbols inside a Region of Interest.
The Control can be used in the following applications:

- Reading of contents of one or more QR symbols inside a ROI;
- Checking of information contents of one or more QR symbols inside a ROI;
- Counting of the number of QR symbols inside a ROI.

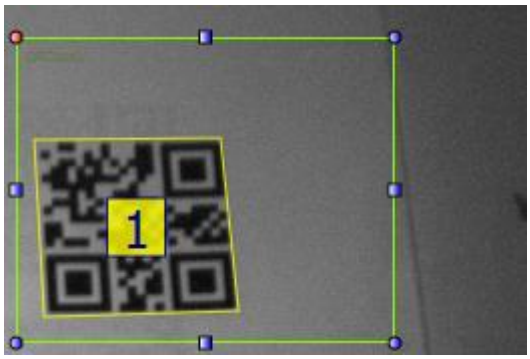
QRCode control selection and positioning

You can select the QRCode Control from the drop-down Control menu in STEP2. The QRCode Control is identified by the following icon:

When you are in STEP 2 the Datamatrix Control allows you to make the following changes:

- ROI Moving: the position of the ROI can be changed in the Reference Image.
- ROI resizing: the width and the height of the ROI can be changed in the Reference Image.

The symbol found is graphically represented inside the ROI with a yellow frame that identifies the vertexes and a rectangle containing the index of the code found, visualized in the centre of code itself. The index, in a decimal format, is displayed to distinguish the Datamatrix symbols found inside the same ROI.



The SUCCESS or FAILURE control result depends on the functioning mode (Control Type) selected. Three functioning modes are accepted by this control:

- **Decoding**: in this mode the QRCode Control decodes all the QR symbols in the ROI. The contents of the symbols decoded is displayed in the **Read Code** area. This Control provides, for each symbol decoded, the contents as well as the corresponding Quality level of each code read, represented by a set of parameters displayed in the **Quality** area. The result sent is SUCCESS if at least one symbol is decoded in the ROI and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure.
- **String Matching**: in this mode the QRCode Control compares the contents of the decoded symbols with one or more character strings listed in the **Expected Code** field. The result sent is SUCCESS if at least one occurrence of each **Expected Code** is found and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure.
- **Counting**: in this mode the QRCode Control verifies that the number of decoded symbols in the ROI belongs to the interval specified in the **Counting Limits** field. The result sent is SUCCESS if the number of symbols decoded is between the Minimum and Maximum values specified and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure.

QR Symbol Characteristics

The QRCode control of DataVS2 decodes the QR symbols with the following characteristics:

- **Coding**: no limits are foreseen relative to the type of coding used for the data inside the symbol. It is possible to read QR codes (model 1 and model 2) and Micro QR codes. In the result string sent by DataVS2 on the Ethernet and Serial channels, the code read is represented in a byte vector format; the byte decoding is made by the client that reads it.

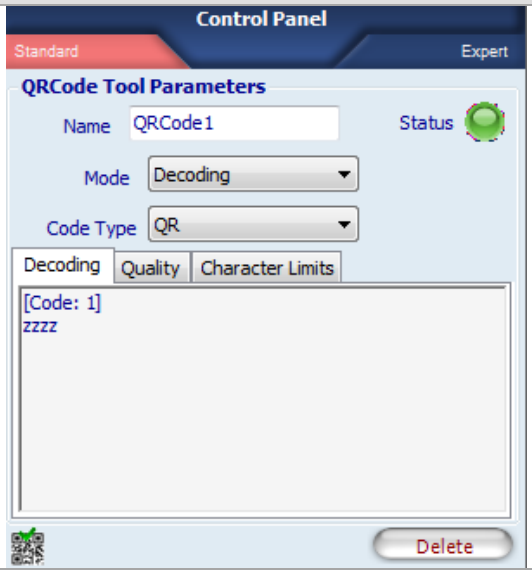
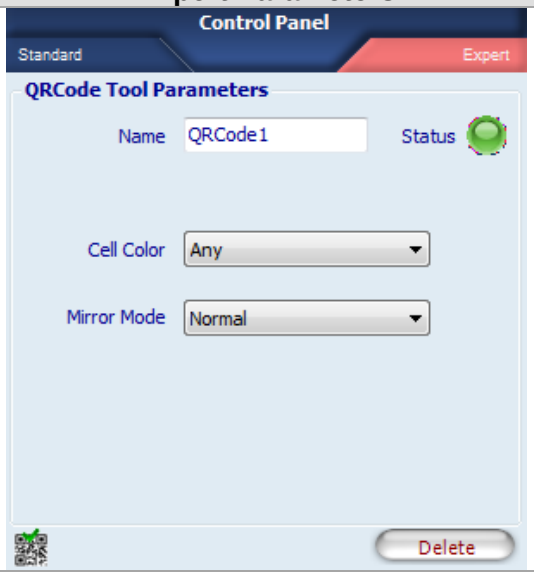
NOTE

DataVS2 GUI decodes all bytes of the code read according to the ASCII table. The contents of a QR symbol that has not been coded in an ASCII format cannot be displayed.

- **Size**: the number of rows and columns composing a QR Code Model 1 symbol that can be decoded using DataVS2 can vary from 21x21 to 73x73, for QR Code Model 2 it can vary from 21x21 to 177x177, increasing in both cases in steps of four modules per side. For the Micro QR symbol instead the number of rows can vary from 11x11 to 17x17, increasing in both cases in steps of two modules per side.
- **Resolution**: the number of pixels per symbol module to decode can vary in the [4, 20] range. This value represents the resolution that characterizes the symbol to decode inside an image.
- **Orientation Angle**: the symbol can have all the possible orientation angles in the ROI.
- **Edges (Quiet Zone):** QR symbols having Edge dimensions equal to 1 module (the dimension of a unit containing information inside the symbol) are accepted.
-

QRCode Control Parameters

The following tables show the QRCode Control parameters:

Standard Parameters	Expert Parameters
	

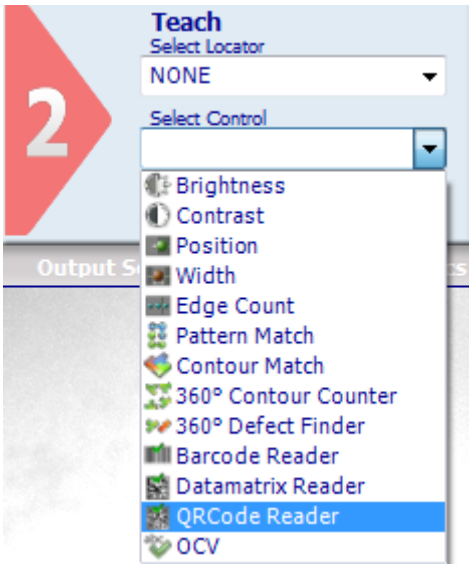
Parameters	Description	Type	Default
Name	The Name assigned to the Control tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	QRCode
Status	Tool that indicates a result: • Green: SUCCESS. The QRCode Control has decoded one or more symbols in the ROI and these correspond to the control filters (Control Type) set. • Red: FAILURE. QRCode Control hasn't decoded any symbols in the ROI or the decoded symbol is incompatible with the filters set.	Standard, Output	
Mode	The **Control Type** parameter selects the functioning mode of the QRCode Control. Three functioning modes can be selected: • **Decoded**: in this mode the QRCode Control decodes all the QR symbols in the ROI. The contents of the symbols decoded is displayed in the **Read Code** area. This Control provides, for each symbol decoded, the contents as well as the corresponding Quality level of each code read, represented by a set of parameters displayed in the **Quality** area. The result sent is SUCCESS if at least one symbol is decoded in the ROI and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure. • **String Control**: in this mode the QRCodeControl compares the contents of the decoded symbols with one or more character strings listed in the **Expected Code** field. The result sent is SUCCESS if at least one occurrence of each **Expected Code** is found and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure. • **Edge Count**: in this mode the QRCode Control verifies that the number of decoded symbols in the ROI belongs to the interval specified in the **Counting Limits** field. The result sent is SUCCESS if the number of symbols decoded is between the Minimum and Maximum values specified and if the Quality value of each decoded symbol is higher or equal to the expected one. On the contrary, the result will be a Failure.	Standard, Input	Decoded
Code Type	The Code Type field defines the type of QR code searched in the ROI. It could be QR or Micro QR	Standard, Input	QR
Decoding	The **Decoding** field displays the contents of each symbol decoded. Each contents is identified by the **[Code i]** prefix where **i** represents the corresponding code index displayed in the Image Panel.	Standard, Output	
Quality	These values represent the general quality of the decoded symbol measured by the QRCode Control. The quality of a QR symbol is represented by the measurement of the following parameters, expressed according to the [0, 1, 2, 3, 4] value scale, where 4 corresponds to the maximum quality value of a code: • **Symbol Contrast**: represents the symbol contrast level expressed as the difference of reflectivity between light and dark parts of the symbol. • **Axial Non Uniformity**: represents the average space difference between two sides of the symbol arranged along the X and Y axes. • **Unused Error Correction**: is the redundancy measurement used for the correction of errors inside	Standard, Output	

Parameters	Description	Type	Default
	the same symbol. • **Decode Grade**: represents the result supplied by the reference decoding algorithm used by the Datamatrix Control. • **Overall Grade**: is equal to the minimum conformity value detected in the quality parameters described above. This value represents the general quality of the symbol and is also graphically displayed next to the **Calculated** label. • The **Expected Value** parameter of the Quality field can be used to set a filter on the expected Quality value. The QRCode control, in each execution, verifies that the **Calculated Value** is higher or equal to the **Expected Value**.		
Expected Code	Allows to specify the expected information contents of one or more symbols to decode. To insert the Expected Code, the user just has to enter the corresponding string of characters in the dedicated area of the control panel. Wild card characters can be used to specify any sequence of characters within the expected string: • "?": use this symbol to specify any character in a known position of the expected string; any decoded symbol corresponding to the position of the jolly symbol will be considered correct. • "*": use this symbol to specify any sequence of characters in a known position of the expected string: any decoded symbol sequence corresponding to the position of the jolly symbol will be considered correct. More than one Expected Code can be specified for each QRCode Control; in this case the Control verifies that each decoded code content matches at least one expected code string. The Teach-in of the Expected Code can be set through the **Auto** selection box; the Control will set the decoded symbol code present in the Reference Image as the Expected Code.	Standard, Input	-
Counting limits	Allows to specify the minimum and maximum number of QRCode symbols that have to be present in the ROI for the Control result to be SUCCESS. These parameters are enabled only when the **Counting** mode is selected: • **Calculated**: represents the number of QR symbols decoded in the ROI. • **Minimum**: represents the minimum number of QR symbols that have to be decoded in the ROI for the Control result to be SUCCESS. • **Maximum**: represents the maximum number of QR symbols that have to be decoded in the ROI for the Control result to be SUCCESS.	Standard, Input	Minimum: 0, Maximum: 100.
Character Limits	It is possible to insert a filter on the number of read characters. The codes which have a number of characters greater than the maximum or less than the minimum are not returned by the tool.	Standard, Input	[0, 3116]
Cell Colour	Represents the colour of the QR symbol to decode: • **Black on White**: black on white background. • **White on Black**: white on black background. • **Any**: the Control in this case tries to decode the symbol according to both the previous options.	Expert, Input	Any
Mirror Mode	Defines the orientation of the cells respect to the **Finder Pattern** of the symbol. The codes can be Normal or Mirrored.	Expert, Input	Normal
Delete	Deletes the Control from the current Inspection.		

Example

- Create an Inspection to verify the contents of a QR symbol.

- Select “QRCode Reader” from the drop-down menu of the available Controls.



- Position the control on the reference image in the IMAGE PANEL: the pointer changes shape as soon as it enters in the work area. Left-click once on any position to create a default rectangular ROI.
- At the same time, the control characteristics and the result of its application on the reference image appear in the control panel.
- Assure that the reference image contains the QR symbol to decode and that it corresponds to the characteristics specified above.
- Set ROI size and position as required. The symbol to be decoded has to be always inside the rectangle visualized.
- The result indicated by STATUS is also highlighted by the ROI border changing colour; the QRCode in the first positioning is configured for a Decoding type control; the result will be SUCCESS if the symbol is decoded If the Control FAILS, check that the ROI has been correctly positioned and that the symbol is clearly visible inside the image, possibly choosing a better reference image of the part to inspect.
- Now change the Control Type parameter selecting the String Control value; then select the Auto box: after this phase, the Control sets the contents of the symbol present in the reference image as expected value. From now onwards, the control will always verify that, from time to time, the contents of the coded symbol are equal to the expected ones.
- It is then possible to test the Inspection configured with images containing different symbols.

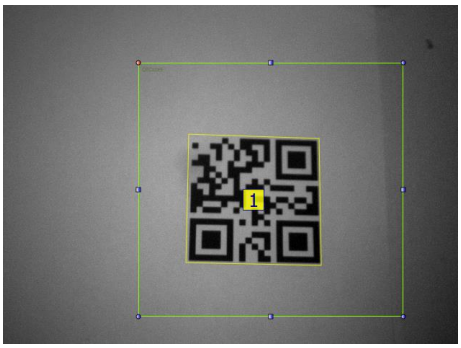


Fig. A: SUCCESS

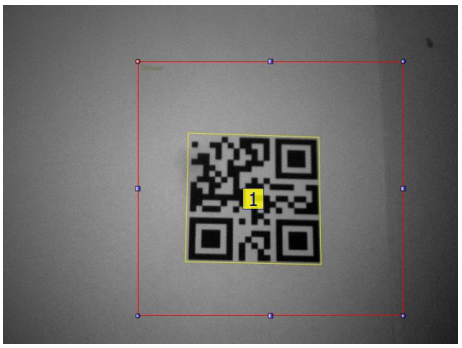


Fig. B: FAILURE

5.4.12 OCV

The OCV (Optical Character Verification) control verifies the presence and correspondence of a character string with respect to a reference image.
The OCV Control can be used in the following applications:

- Verify the presence of a string of characters in a ROI.
- Verify the quality of a string of characters in a ROI.

The OCV control cannot be used to decode a string of characters.

OCV Control selection and enabling

You can select the OCV Control from the drop-down Control menu in STEP2. The OCV Control is identified by the following icon:

When you are in STEP 2 the Barcode Control allows you to make the following changes:

- ROI Moving: the position of the ROI can be changed in the Reference Image.
- ROI Resizing: the width and the height of the ROI can be changed in the Reference Image.
- ROI Rotation: the orientation of the ROI can be changed in the Reference Image.

After having positioned the ROI, the image pixels corresponding to the characters found are graphically displayed in green. A skeleton is indicated for each character, represented by a line that follows the shape.

Some useful hints that can help you choose a good target object image:

- The reference image has to be in focus and have a good contrast and brightness level; you can adjust image brightness and contrast by changing the image acquisition Exposure Time and Gain in STEP1.

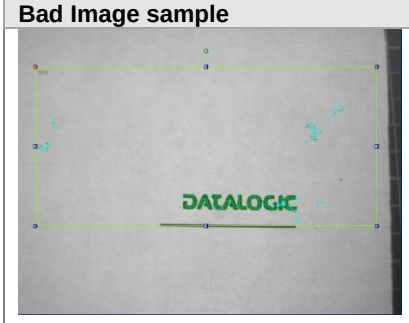
- The target object has to be completely included inside the template ROI borders; adjust TEMPLATE ROI dimensions and position in order to include only the object to be searched.

- The target object texture has to be recognizable and in contrast with its background; assure that the reference image does not contain ambiguous objects or characteristics having same texture of the target object. Only the target object should be present in the reference image while the remaining part of the image (the background) should be uniform.

- Verify that all the characters of the target object are correctly coloured and that the skeleton highlighted for each character corresponds to the shape of the character itself. If possible, assure that the space between each character and the adjacent one is constant inside the string.

The following table shows some examples of bad target object configuration:

Bad Image sample	Description

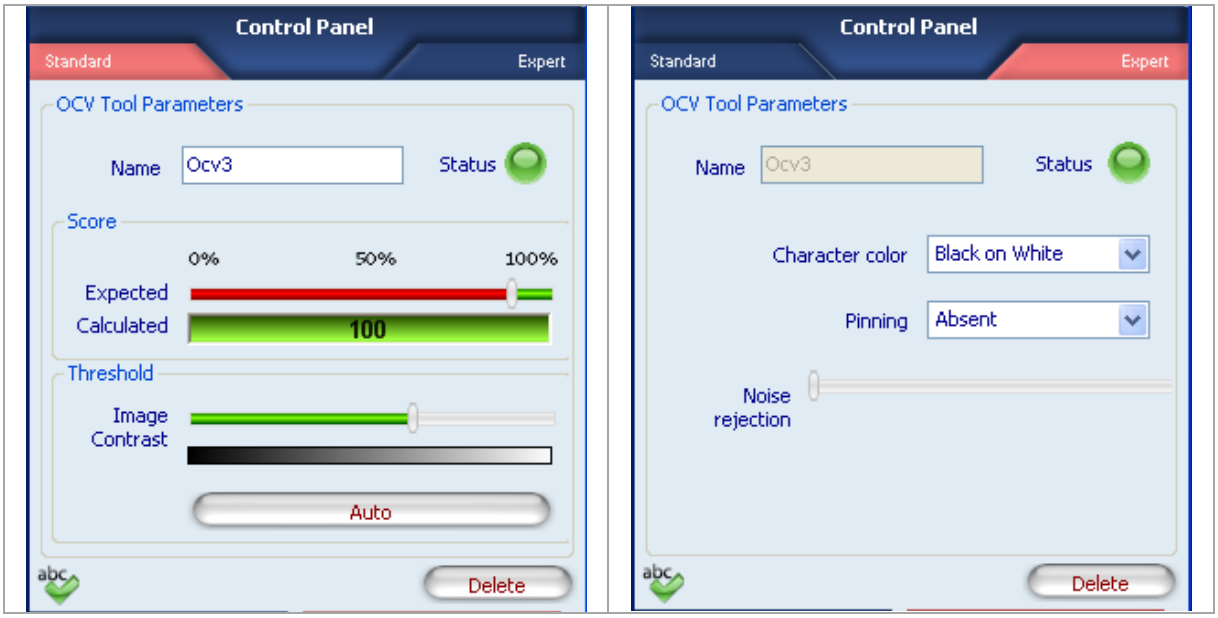
Bad Image sample	Description
	The target object is too small with respect to the ROI dimensions and the characters cannot be adequately distinguished.
	The target object is not completely inside the borders of the template ROI. The dimensions of the template ROI should be adjusted in this case.
	The skeleton drawn on the target object doesn't coincide with the character shape to detect. We suggest to acquire a better reference image of the target object.

The result of the Teach phase of the target string is represented by the colour assumed by the border of the ROI: green in case of SUCCESS, red in case of FAILURE. In the latter, we suggest to re-position the target so as to perfectly comply with the indications given above.

OCV Control Parameters

The following table shows the OCV Control parameters:

Standard Parameters	Expert Parameters

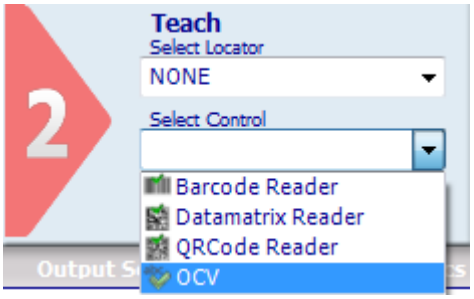


Parameters	Description	Type	Default
Name	The Name assigned to the Control tool. The maximum length accepted is 256 UTF-8 characters. **Note**: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	OCV
Status	Tool that indicates a result: • Green: SUCCESS. The OCV Control has detected a string of characters in the ROI having a calculated score value higher or equal to the expected one. • Red: FAILURE. the control has not detected any string of characters in the ROI or the score calculated for the character string is lower than the expected one.	Standard, Output	
Calculated	Represents the success percentage that identifies the degree of similarity between the string of characters found and the reference image.	Standard, Output	
Expected	Represents the threshold used to calculate the result. It indicates how closely the string of characters in the ROI must resemble the reference image to be considered valid. 100% indicates a perfect match, 0% indicates no match found.	Standard, Input	90%
Image Contrast	Gray scale threshold. By changing this value you select which pixels of the ROI are part of the character string and which are not. It is compared to the value detected on the acquired image: if the latter value is lower than the threshold, the point is not considered part of the object, otherwise the point is considered part of the object We recommend to set the threshold to a gray level value near the average gray level of the background. **Note**: the value is automatically set by the DataVS2 GUI software in the first configuration. This value is strictly related to the gray level of the pixels inside the ROI, each time you move the ROI in STEP 2 the binarization threshold is automatically adjusted by the DataVS2 GUI. You can always ask the DataVS2 GUI to automatically calculate the threshold by pressing the “AUTO” button.	Standard, Input	AUTO
Auto	Forces the automatic setup of the binarization threshold; it is useful to restore the first configuration after changing	Standard, Input	

Parameters	Description	Type	Default
	the binarization threshold value.		
Character Colour	Specifies the type of characters belonging to the string to detect; two are the possible values: • **Black on White**: all the characters of the string to detect must be dark (having the same colour) on a light coloured background. • **White on Black**: all the characters of the string to detect must be light coloured having the same colour) on a dark coloured background.	Expert, Input	Black on White
Dotpeen	Denotes the print typology that characterizes the character string to detect. It is expressed through the percentage of dots visible to the human eye inside each character. Possible values: • Absent • Low • High • Evident	Expert, Input	Absent
Noise Rejection	Reduces the effect that noisy pixels will have on the variation detection. The higher the value, the stronger the brightness variation required.	Expert, Input	30%
Delete	Deletes the Control from the current Inspection.		

Example

- Create an Inspection to verify the contents of a string of characters.
- Select “OCV” from the drop-down menu of the available Controls.



- Position the control on the reference image in the IMAGE PANEL: the pointer changes shape as soon as it enters in the work area. Left-click once on any position to create a default rectangular ROI.
- At the same time, the control characteristics and the result of its application on the reference image appear in the control panel.
- Assure that the reference image contains the template character string and that it corresponds to the characteristics specified above.
- Set ROI size and position as required. The symbol to be decoded has to be always inside the rectangle visualized.

- The result indicated by STATUS is also highlighted by the ROI border changing colour; the result of the Teach phase of the master object will be SUCCESS if the characters are acquired correctly and the score is equal to 100%. If the Control FAILS, that the ROI has been correctly positioned and that the symbol is clearly visible inside the image, possibly choosing a better reference image of the part to inspect.

- Completed the Teach-in phase of the sample (Step2), it is now possible to test the efficiency of the configuration in Step3 and acquire good and bad images of the part.

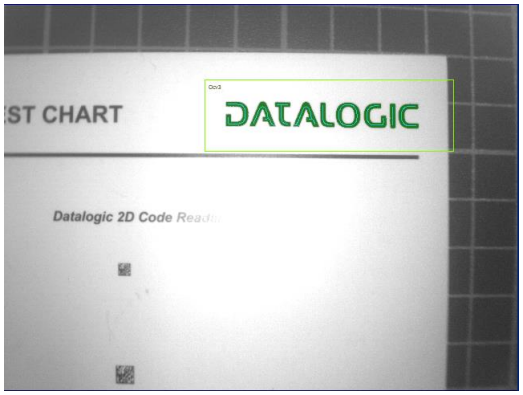


Fig. A: SUCCESS

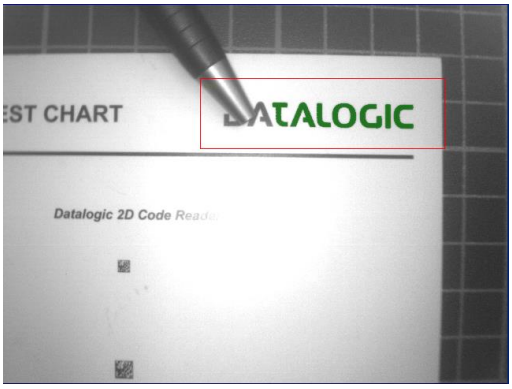


Fig. B: FAILURE

5.4.13 360° Contour Counter

(only available in DataVS2 PRO Model)

The 360° Contour Counter Control is able to detect the position and orientation of one or more similar objects inside a well-defined image area. The number of objects found is included amongst the results returned at the end of the processing stage of each image.

360° Contour Counter Control: selection and positioning

You can select the 360° Contour Counter Control from the Controls pull-down menu in STEP 2. The 360° Contour Counter Control is identified by the following icon:

The 360° Contour Counter Control is characterized by two different ROIs, having different scopes:

- SEARCH AREA: the outer rectangle represents the portion of the image where the Locator will search for the target object.
- TEMPLATE ROI: the inner rectangle represents the object template to be searched in the region delimited by the SEARCH AREA. The TEMPLATE ROI cannot be positioned outside the SEARCH AREA.

When you are in STEP 2, the 360° Contour Counter Control ROI allows the following variations:

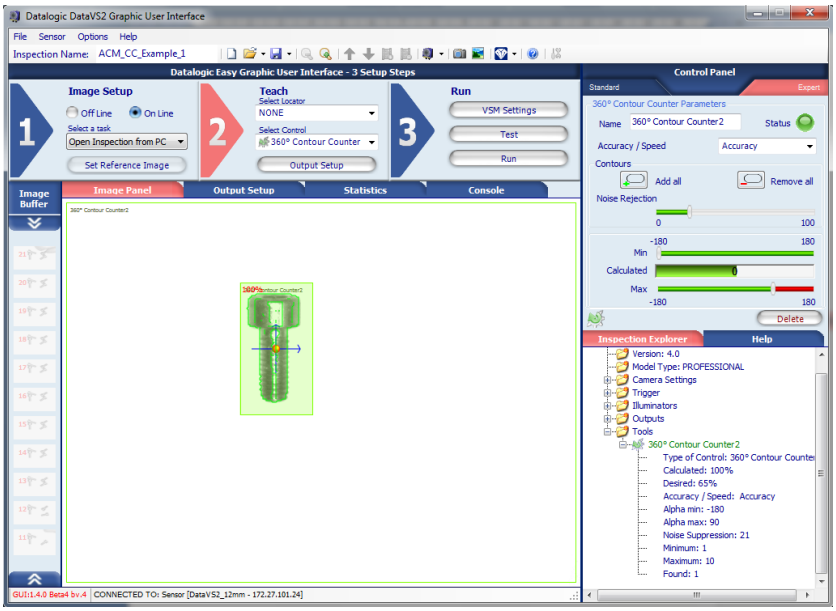
- Move the SEARCH AREA and the TEMPLATE ROI: the position of the ROI can be changed inside the Reference Image.
- SEARCH AREA and TEMPLATE ROI resizing: the width and the height of the ROI can be changed inside the Reference Image

During the setup process, you need to define the target object to be identified. The system memorizes the characteristics of the object to detect, represented by the object contours.
An Edge in a digital image is a geometrical area of a group of pixel characterized by the same gray level and have adjacent pixel elements with a lower or higher gray level.

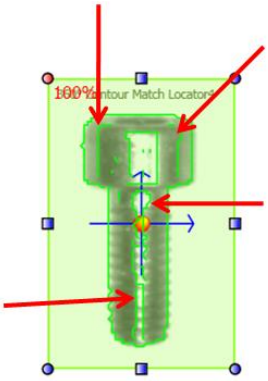
360° Contour Counter Control: contour configuration

When the 360° Contour Counter Control is configured, the object to detect must be characterized by clear contours, very evident inside the reference image.
For this reason, during the configuration of the TEMPLATE ROI, the group of contours found can be modified by selecting the good ones and deleting the less representative contours of the object from the template.
The fundamental steps of the template configuration are listed below.

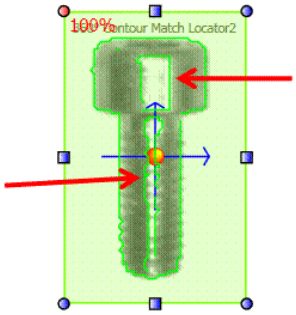
1. Ensure that the reference image contains only one object to use as target.
2. Verify the contours acquired on the template: after positioning the TEMPLATE ROI on the reference image, the system highlights the contours found inside it. These are represented by green curves visualized inside the TEMPLATE ROI. To avoid acquiring contours not belonging to the template ensure that the TEMPLATE ROI contains only the desired target. If the reference image presents areas with variable brightness or reflections generated by the object itself, meaningless contours can be detected. In this case remove noise. The following image shows the configuration phase of a 360° Contour Counter Control for the detection of a metal screw.



The red arrows in the following figure highlight meaningless contours, corresponding to the light reflected from the screw surface. These will be removed from the template.

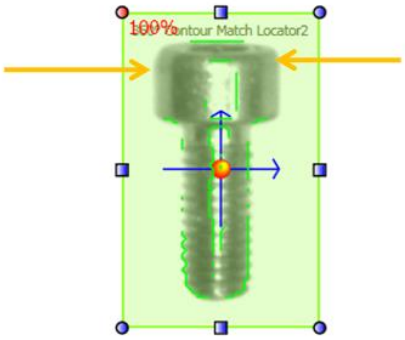


3. Remove noise: contours generated by weaker reflections can be removed using the “Noise Removal” parameter in the Expert panel of the 360° Contour Counter Control parameters. In general, the higher the value of this parameter, the lower the number of contours found inside the TEMPLATE ROI. The following figure shows the effect of this parameter on the example: spurious weaker contours have been eliminated.

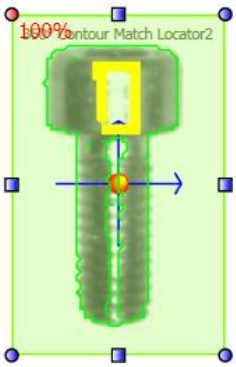


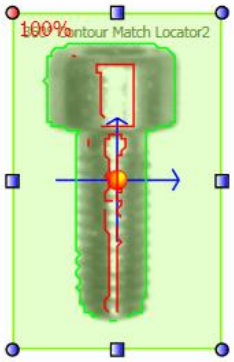
NOTE

When contours are removed, pay attention not to remove meaningful object contours. The following figure shows the effect of an excessive increase of the "Noise Removal" threshold: some meaningful object contours have been eliminated.



4. Manual removal of meaningless contours: if at end of the noise removal, some meaningless contours are still present, the user can remove them manually. To proceed manually, the user must select each contour to remove with the left mouse button. When aimed, the contour to remove will be highlighted in yellow. When selected, the contour to remove will be represented in red.





NOTE

If some contours have been erroneously eliminated, they can be reintroduced selecting them again with the mouse.

5. Selection of desired resemblance and search angle: once ended the contour configuration, the user can select the resemblance degree using the "Expected" parameter.

NOTE

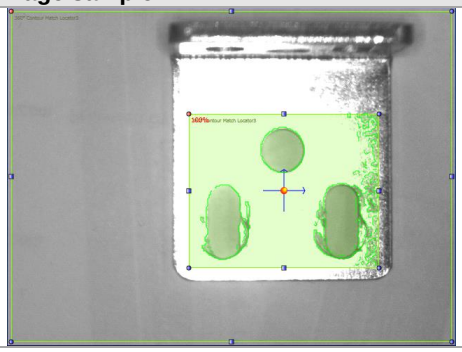
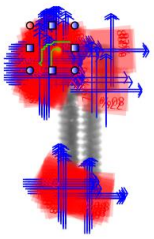
In order to have a reliable and robust behaviour of the 360° Contour Counter Control we recommend you to configure the TEMPLATE ROI using a well contrasted reference image of the target object.

Some useful hints that can help you choose a good target object image:

- The reference image has to be in focus and have a good contrast and brightness level; you can adjust image brightness and contrast by changing the image acquisition Exposure Time and Gain in STEP1. Assure that the highlighted contours (orange curves) are placed on the target object to recognize and not on its background.
- The target object has to be completely included inside the template ROI borders; adjust TEMPLATE ROI dimensions and position in order to include only the object to be searched. Assure that the highlighted corners (red squared dots) are located uniformly on the target object. Assure at least 4 corners are present on the target object.
- The target object texture should be recognizable and in contrast with its background; assure the reference image doesn't contain ambiguous objects or features having the same texture of the target object. Only the target object should be present in the reference image while the remaining part of the image (the background) should be uniform. Assure that the highlighted contours are placed on the target object to recognize and not on its background and assure that all corners are on the target object.

The DataVS2 GUI software helps you in defining a good quality TEMPLATE ROI: a popup message will be shown whenever the chosen target object is not good (see picture below).
When this message is shown, please try to adjust the TEMPLATE ROI by the mean of resizing or moving it. If nothing changes, we suggest you to choose a different reference image.

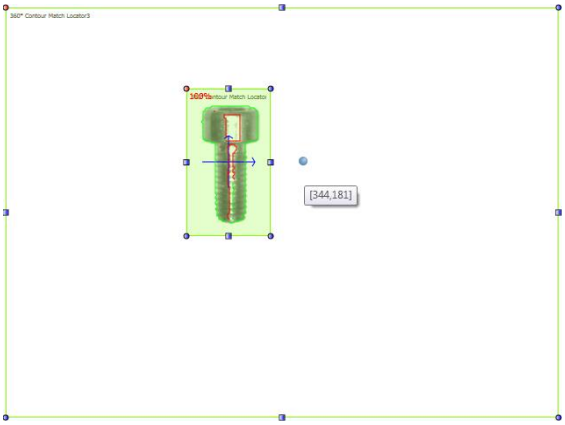
In the following table you will find some examples of a bad object contour configuration:

Bad Image sample	Description
	The reference image does not present homogeneous lighting; the light reflections produced by the metal surface of the object can generate instability of the object contours. In this case we suggest to change the image acquisition parameters (Exposure Time and Gain) until a more uniform image is obtained.
	The TEMPLATE ROI is not configured correctly; the highlighted object characteristics (the contours) are not sufficient to detect it, as they are not unique. When an object is characterized by repetitive geometrical elements (elliptic holes in the example) we suggest to include them all in the template.

360° Contour Counter Control: configuration of object acquisition point

The object acquisition point represents the position of the object respect to an absolute reference system centred in the vertex on the top left side of the image. It is made up of a group of Cartesian coordinates (X, Y) corresponding to the barycentre of the green rectangle that identifies the object detected in the image.

During the configuration of the 360° Contour Match locator, an acquisition point different from the default one can be defined. The user has to select the red anchor, visualized in the ROI center, with the mouse and drag it in the desired position. The new position defined for the red anchor represents the object position with respect to the green rectangle.

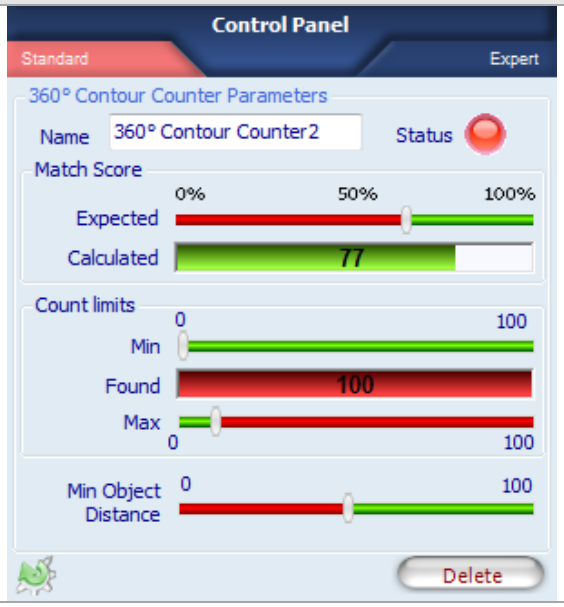
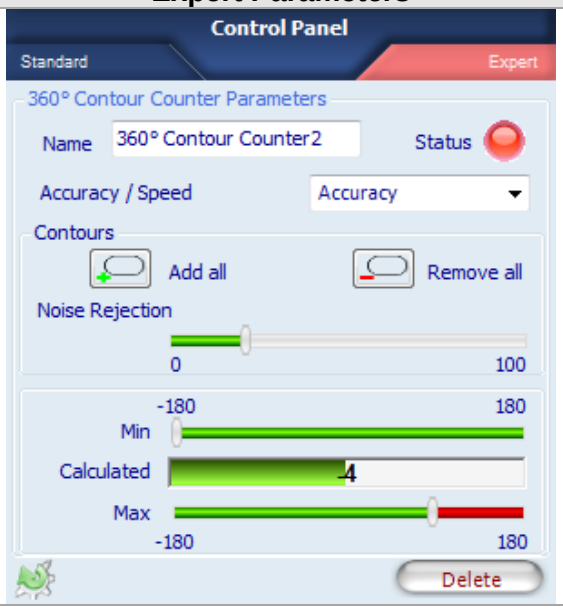


NOTE

The sensor returns as position found the one corresponding to the acquisition point.

Parameters of the 360° Contour Counter Control

The following image and table represent the 360° Contour Counter Control parameters.

Standard Parameters	Expert Parameters
	

Parameters	Description	Type	Default
Name	The Name assigned to the Locator Tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	360° Contour Match Locator
Status	A tool result indicator . • Green: SUCCESS. The Pattern Match Locator has found the target pattern with a score higher that the expected one. • Red: FAILURE. Pattern Match Locator hasn't found the target pattern or the score of the found pattern is lower than the expected one.	Standard, Output	
Calculated	The score representing the calculated degree of similarity of the current found object to the reference target one.	Standard, Output	
Expected	Threshold used for the result evaluation. Indicates how much the object found in the SEARCH AREA must resemble the reference object to be considered valid. 100% means perfect match, 0% means no match found.	Standard, Input	75%
Counting limits	Defines the number of expected objects inside an image. • **Found**: the number of found occurrences of the target object inside the image having a matching score higher than the expected one. • **Minimum**: represents the minimum number of occurrences of the object to find inside an image until the result can be SUCCESS. If the number of objects found is below this value, the control is FAIL • **Maximum**: represents the maximum number of occurrences of the object to find inside an image until the result can be SUCCESS. If the number of objects found is over this value, the control is FAIL	Standard, Input	Minimum: 0 Maximum: 100
Minimum distance	Defines the minimum distance (expressed as a percentage of the size of the model) that must be present between two objects identified as similar inside an image so that they can be considered valid.	Standard, Input	50
Search angle	Allows to set the limits on the orientation angle used to search the object inside the SEARCH AREA. • **Calculated**: the orientation degree of the found target object. ***Note***: the orientation degree of the found object is always referred to the interval [-180°, +180°]. • **Minimum**: The minimum orientation degree that the found object must have in order to return SUCCESS. If found object has an orientation degree lower than minimum expected the tool will fail. • **Maximum**: The maximum orientation degree that the found object must have in order to return SUCCESS. If found object has an orientation degree higher than the maximum expected the tool will FAIL.	Standard, Input	Minimum: -90° Maximum: 90°
Noise Rejection	Use this parameter to reduce the number of contours not belonging to the target object or not related to the elements characterizing the target object. This is useful when the image is not well contrasted or present some noise (shadows and dirty).	Expert, Input	20%
Add all	Press this button to include all the contours found in the reference template. After pressing this button, all the contours in the image are visualized in green.	Expert, Input	
Delete all	Press this button to eliminate all the contours found in the reference template. After pressing this button, all the contours in the image are visualized in red.	Expert, Input	

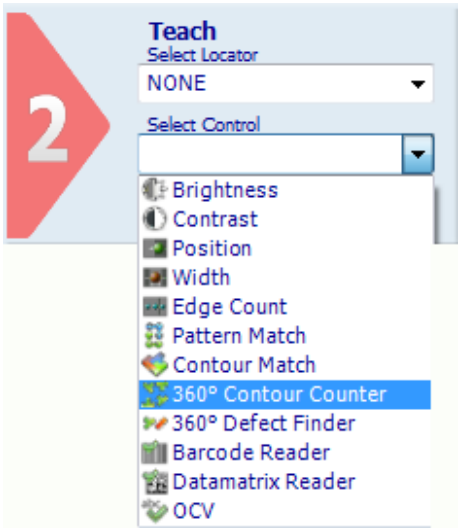
Parameters	Description	Type	Default
Mode (Accuracy / Speed)	Represents the level of accuracy used by the 360° Contour Match Tool in searching the target. In Accuracy mode the Tool performs a two-phase processing: the first one to locate the object, the second one to accurately define its match score. In Speed mode the Tool only performs the first phase of the processing; the match score in this case is calculated in the first phase. In Speed mode the Tool is faster. ***Note***: we recommend to set this parameter to Accuracy when you have to perform quality controls or when the Reference Image presents a non-uniform background; Speed mode has to be used for location purposes only and when the target object is well defined in comparison with its background. ***Note***: the Tool automatically sets the best value for Accuracy/Speed parameter depending on the Reference Image characteristics.	Expert, Input	Accuracy
Delete	Deletes the Locator from the current Inspection		

To quicken the 360° Contour Counter Control execution time you can act on two different parameters:

- Reducing the SEARCH AREA dimensions: if you are sure the object to be found is always in a restricted area of the image, we recommend you to reduce SEARCH AREA dimension in order to cover only the portion of the image where you want the object to be searched. This will reduce time needed to locate the object (as the portion of the image where searching for the target pattern is smaller) and will decrease the risk of errors due to noise inside the image.
- Setting the Accuracy /Speed parameter to Speed: reduce the precision level requested during the search phase will reduce the time necessary to calculate the score.

Example

- Add a "360° Contour Counter" Control to current inspection.
- Select "360° Contour Counter" from the pull-down menu.



- Position the control on the reference image in the IMAGE PANEL: the pointer changes shape as soon as it enters the work area. Left-click once at any position to create a default rectangular ROI.

- At the same time, the control features and the result of its application to the reference image appear in the control panel.
- The ROI and contours must be configured according to the requirements following the modes explained previously.

- The result indicated by STATUS is also highlighted by the ROI border changing colour.
- As soon as values change, result is calculated without delay.

NOTE

If the object to detect features a geometry that causes a variation in the light reflected according to the position and orientation angle of the object, we suggest to set the "Expected" threshold to a value not over 70%.

- Hereinafter, the Figures A and B show a working example of the Contour Counter Control 360°. In this particular case, the 360° Contour Counter Control searches objects similar to the screw inside the image and returns the position and number of occurrences found.

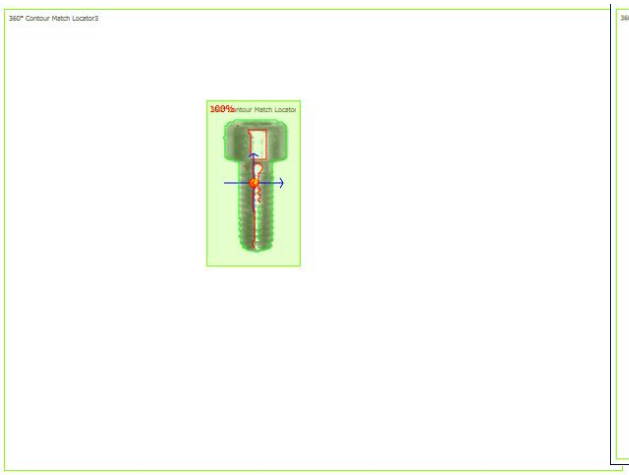


Fig. A Control Contour Counter: Reference Image

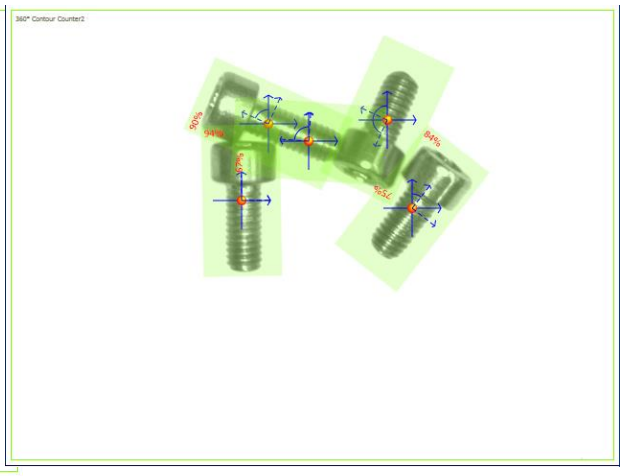


Fig. B Contour Counter Control: Effect of an inspection on an image containing more than one occurrence of the master object

5.4.14 360° Defect Finder

(only available in DataVS2 PRO Model)

The 360° Defect Finder Control is able to detect the presence of defects, measuring the difference between the contour of a master object and the contour of the object currently analysed.

360° Defect Finder Control: selection and positioning

You can select the 360° Defect Finder Control from the Controls pull-down menu in STEP 2. The 360° Defect Finder Control is identified by the following icon:

The 360° Defect Finder Control is characterized by two different ROIs, having different scopes:

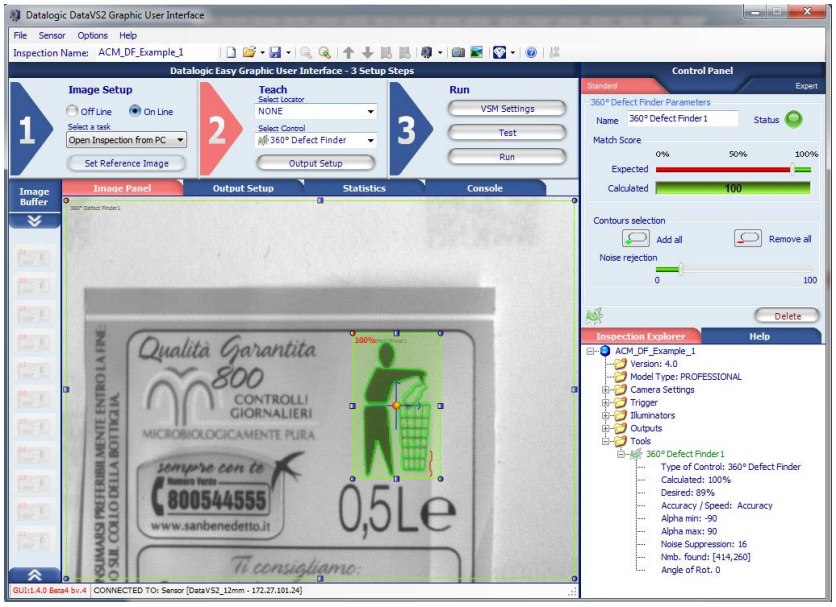
- SEARCH AREA: the outer rectangle represents the portion of the image where the Locator will search for the target object.
- TEMPLATE ROI: the inner rectangle represents the object template to be searched in the region delimited by the SEARCH AREA. The TEMPLATE ROI cannot be positioned outside the SEARCH AREA.

When you are in STEP 2, the 360° Defect Finder Control ROI allows the following variations:
Move the SEARCH AREA and the TEMPLATE ROI: the position of the ROI can be changed inside the Reference Image.
SEARCH AREA and TEMPLATE ROI resizing: the width and the height of the ROI can be changed inside the Reference Image
During the setup process, you need to define the target object to be identified. The system memorizes the characteristics of the object to detect, represented by the object contours.
An Edge in a digital image is a geometrical area of a group of pixel characterized by the same gray level and have adjacent pixel elements with a lower or higher gray level.

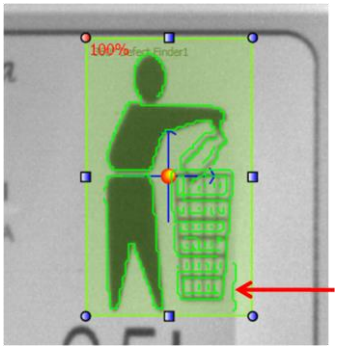
360° Defect Finder Control: contour configuration

When the 360° Defect Finder Control is configured, the object to detect must be characterized by clear contours, very evident inside the reference image.
For this reason, during the configuration of the TEMPLATE ROI, the group of contours found can be modified by selecting the good ones and deleting the less representative contours of the object from the template. The fundamental steps of the template configuration are listed below.

1. Ensure that the target object characteristics are highlighted in the reference image to detect the defects during Inspection execution.
2. Verify the contours acquired on the template: after positioning the TEMPLATE ROI on the reference image, the system highlights the contours found inside it. These are represented by green curves visualized inside the TEMPLATE ROI. To avoid acquiring contours not belonging to the template ensure that the TEMPLATE ROI contains only the desired target. If the reference image presents areas with variable brightness or reflections generated by the object itself, meaningless contours can be detected. In this case remove noise. The following image shows the configuration phase of a 360° Defect Finder Control for the detection of a logo.

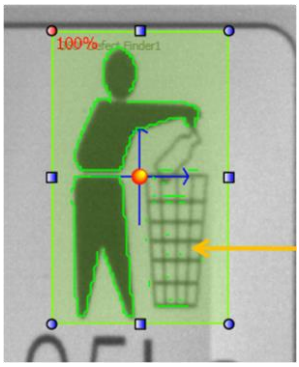


The red arrows in the following figure highlight meaningless contours, corresponding to the light reflected from the screw surface. These will be removed from the template.

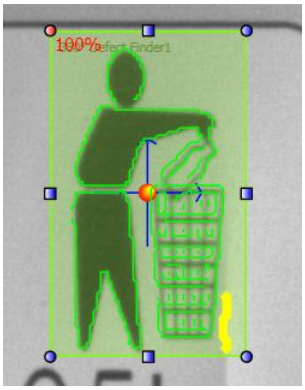


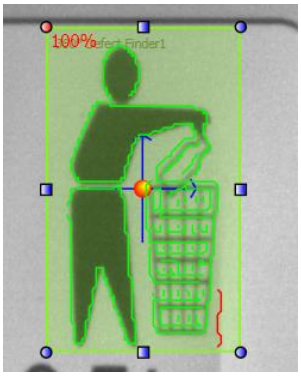
3. Noise removal: contours generated by weaker reflections can be removed using the "Noise Removal" parameter in the Expert panel of the 360° Defect Finder Control parameters. In general, the higher the value of this parameter, the lower the number of contours found inside the TEMPLATE ROI.

NOTE *When contours are removed, pay attention not to remove meaningful object contours. The following figure shows the effect of an excessive increase of the "Noise Removal" threshold: some meaningful object contours have been eliminated.*



4. Manual removal of meaningless contours: if at end of the noise removal, some meaningless contours are still present, the user can remove them manually. To proceed manually, the user must select each contour to remove with the left mouse button. When aimed, the contour to remove will be highlighted in yellow. When selected, the contour to remove will be represented in red.





NOTE

If some contours have been erroneously eliminated, they can be reintroduced selecting them again with the mouse.

5. Selection of desired resemblance and search angle: once ended the contour configuration, the user can select the resemblance degree using the “Expected” parameter.

NOTE

In order to have a reliable and robust behaviour of the 360° Defect Finder Control we recommend you to configure the TEMPLATE ROI using a well contrasted reference image of the target object.

Some useful hints that can help you choose a good target object image:

- The reference image has to be in focus and have a good contrast and brightness level; you can adjust image brightness and contrast by changing the image acquisition Exposure Time and Gain in STEP1. Assure that the highlighted contours (orange curves) are placed on the target object to recognize and not on its background.
- The target object has to be completely included inside the template ROI borders; adjust TEMPLATE ROI dimensions and position in order to include only the object to be searched. Assure that the highlighted corners (red squared dots) are located uniformly on the target object. Assure at least 4 corners are present on the target object.
- The target object texture should be recognizable and in contrast with its background; assure the reference image doesn’t contain ambiguous objects or features having the same texture of the target object. Only the target object should be present in the reference image while the remaining part of the image (the background) should be uniform. Assure that the highlighted contours are placed on the target object to recognize and not on its background and assure that all corners are on the target object.

The DataVS2 GUI software helps you in defining a good quality TEMPLATE ROI: a popup message will be shown whenever the chosen target object is not good (see picture below).
When this message is shown, please try to adjust the TEMPLATE ROI by the mean of resizing or moving it. If nothing changes, we suggest you to choose a different reference image.

In the following table you will find some examples of a bad object contour configuration:

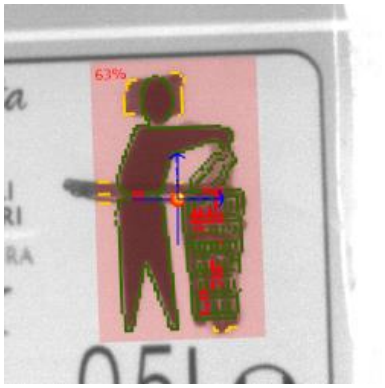
Bad Image sample	Description
	The reference image does not present homogeneous lighting; the light reflections produced by the metal surface of the object can generate instability of the object contours. In this case we suggest to change the image acquisition parameters (Exposure Time and Gain) until a more uniform image is obtained.
	The TEMPLATE ROI is not configured correctly. The object characteristics highlighted (the contours) are not sufficient to detect it, as not completely contained inside the TEMPLATE ROI.

360° Defect Finder Control: defect display

After the configuration phase of the master object, the user can then verify the good quality of the configuration carried out passing to STEP 3, Test.
The 360° Defect Finder Control measures the defect index of the object currently analysed detecting both the absence of contours present in the reference image and the presence of contours in excess, that is not present in the reference image.

All the contours absent from the analysed object are highlighted in red, as shown in the following example:

All the contours in excess respect to those acquired from the reference image are highlighted in yellow:

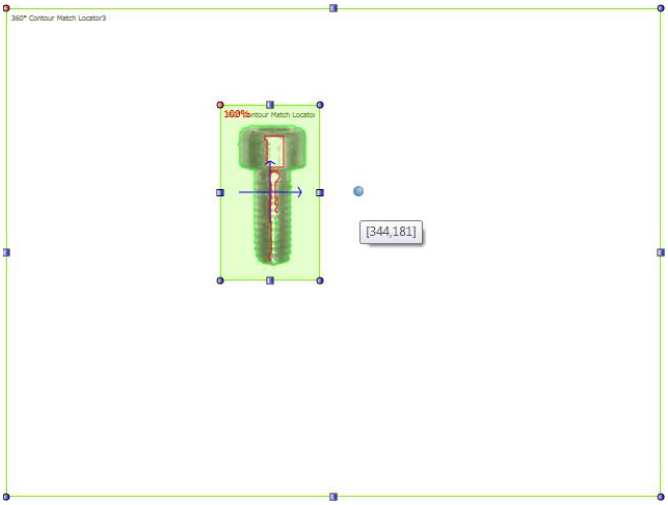


If the object to analyze shows a high defect index, it may not be detected by the 360° Defect Finder Control. In this case, only the SEARCH AREA is displayed in red on the processed image.

360° Defect Finder Control: configuration of object acquisition point

The object acquisition point represents the position of the object respect to an absolute reference system centred in the vertex on the top left side of the image. It is made up of a group of Cartesian coordinates (X, Y) corresponding to the barycentre of the green rectangle that identifies the object detected in the image.

During the configuration of the 360° Defect Finder locator, an acquisition point different from the default one can be defined. The user has to select the red anchor, shown in the ROI center, with the mouse and drag it in the desired position. The new position defined for the red anchor represents the object position with respect to the green rectangle.

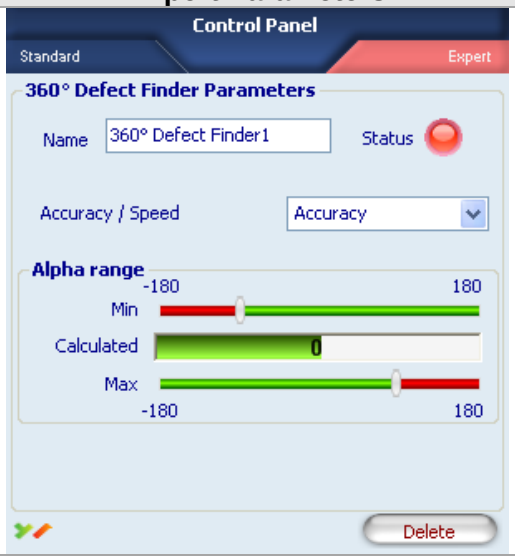


NOTE

The sensor returns as position found the one corresponding to the acquisition point.

Parameters of the 360° Defect Finder Control

The following image and table represent the 360° Defect Finder Control parameters.

Standard Parameters	Expert Parameters
	

Parameters	Description	Type	Default
Name	The Name assigned to the Locator Tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	360° Contour Match Locator
Status	A tool result indicator . • Green: SUCCESS. The Pattern Match Locator has found the target pattern with a score higher that the expected one. • Red: FAILURE. Pattern Match Locator hasn't found the target pattern or the score of the found pattern is lower than the expected one.	Standard, Output	
Calculated	The score representing the calculated degree of similarity of the current found object to the reference target one.	Standard, Output	
Expected	Threshold used for the result evaluation. Indicates the maximum defect index that the object found can have for the control result to be SUCCESS. 100% indicates the absence of defects.	Standard, Input	75%
Noise Rejection	Use this parameter to reduce the number of contours not belonging to the target object or not related to the elements characterizing the target object. This is useful when the image is not well contrasted or present some noise (shadows and dirty).	Standard, Input	20%
Add all	Press this button to include all the contours found in the reference template. After pressing this button, all the contours in the image are visualized in green.	Standard, Input	
Delete all	Press this button to eliminate all the contours found in the reference template. After pressing this button, all the contours in the image are visualized in red.	Standard, Input	
Considers additional contours	When selected the score is calculated by considering also the contours in excess; when it is not selected the score is calculated by considering only the missing contours.	Standard, Input	Selected

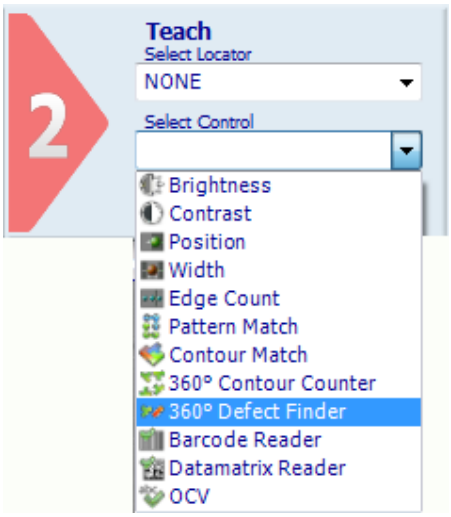
Parameters	Description	Type	Default
Search angle	Allows to set the limits on the orientation angle used to search the object inside the SEARCH AREA. • **Calculated**: the orientation degree of the found target object. ***Note***: the orientation degree of the found object is always referred to the interval [-180°, +180°]. • **Minimum**: The minimum orientation degree that the found object must have in order to return SUCCESS. If found object has an orientation degree lower than minimum expected the tool will fail. • **Maximum**: The maximum orientation degree that the found object must have in order to return SUCCESS. If found object has an orientation degree higher than the maximum expected the tool will FAIL.	Expert, Input	Minimum: -90° Maximum: 90°
Mode (Accuracy / Speed)	Represents the level of accuracy used by the 360° Defect Finder Control in searching the target. In Accuracy mode the Tool performs a two-phase processing: the first one to locate the object, the second one to accurately define its match score. In Speed mode the Tool only performs the first phase of the processing; the match score in this case is calculated in the first phase. In Speed mode the Tool is faster. ***Note***: we recommend to set this parameter to Accuracy when you have to perform quality controls or when the Reference Image presents a non-uniform background; Speed mode has to be used for location purposes only and when the target object is well defined in comparison with its background. ***Note***: the Tool automatically sets the best value for Accuracy/Speed parameter depending on the Reference Image characteristics.	Expert, Input	Accuracy
Delete	Deletes the Locator from the current Inspection		

To quicken the 360° Defect Finder Control execution time you can act on two different parameters:

- Reducing the SEARCH AREA dimensions: if you are sure the object to be found is always in a restricted area of the image, we recommend you to reduce SEARCH AREA dimension in order to cover only the portion of the image where you want the object to be searched. This will reduce time needed to locate the object (as the portion of the image where searching for the target pattern is smaller) and will decrease the risk of errors due to noise inside the image.
- Setting the Accuracy /Speed parameter to Speed: reduce the precision level requested during the search phase will reduce the time necessary to calculate the score.

Example

- Add a "360° Defect Finder" Control to current inspection.
- Select "360° Contour Counter" from the pull-down menu:



- Position the control on the reference image in the IMAGE PANEL: the pointer changes shape as soon as it enters the work area. Left-click once at any position to create a default rectangular ROI.
- At the same time, the control features and the result of its application to the reference image appear in the control panel.
- The ROI and contours must be configured according to the requirements following the modes explained previously.
- The result indicated by STATUS is also highlighted by the ROI border changing colour.
- As soon as values change, result is calculated without delay.
- Hereinafter, the Figures A and B show a working example of the 360° Defect Finder Control. In this particular case, the 360° Defect Finder Control detects defects in a label logo.

Fig. A 360° Defect Finder Control: Reference Image

Fig. B 360° Defect Finder Control: Inspection effect on a faulty image

5.5 LOGICAL TOOLS

(not available in DataVS2 OBJ Model)

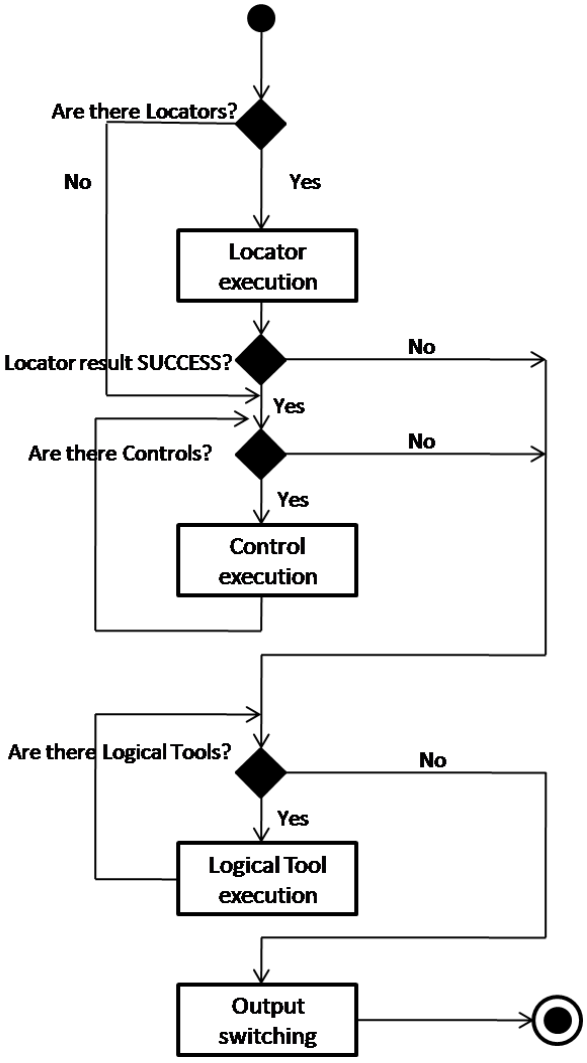
Logical Tools perform logic operations on one or more logic inputs and produce a single logic output. In the sensor application domain, the Logical Tool inputs are logical results of the Inspection Controls. Logical Tools are useful to configure a logical network between Inspection Controls and sensor outputs: in this way it is possible to connect the result of a subset of Inspection Controls to a single output of the sensor.

Logical Tools are positioned graphically on the OUTPUT SETUP PANEL using the mouse. When you insert a Logical Tool, the CONTROL PANEL displays the set of possible inputs that can be connected to the Logical Tool; a Logical Tool input can only be an inspection Control or a Logical Tool already inserted in the Inspection.

The execution of one of these Tools provides a single type of result, **Status**, representing the logical result calculated during the Tool processing. Two are the possible values for the status of a Logical Tool:

- SUCCESS: The tool has supplied a positive result, a Boolean TRUE.
- FAILURE: The tool has supplied a negative result, a Boolean FALSE.

The following picture displays the flowchart of the Inspection execution when Logical Tools are present in the Inspection. Logical Tools are always executed after Controls and Locators execution, regardless of the Controls logical result, SUCCESS or FAILURE.

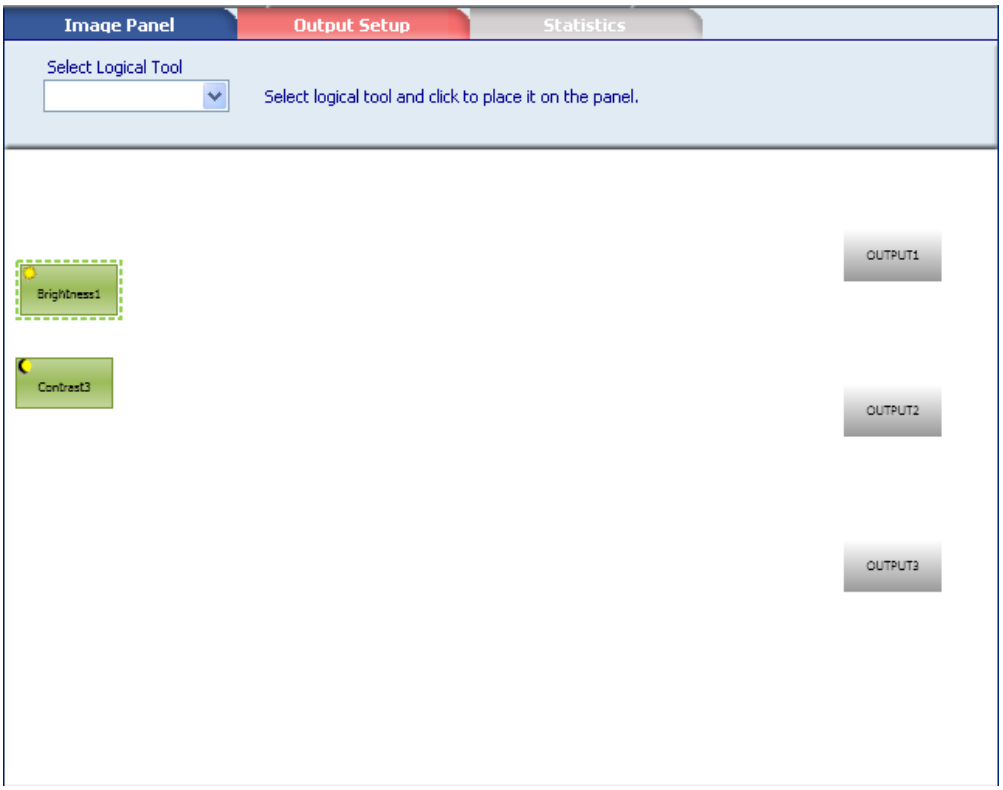


The following table list the available Logical Tools in DataVS2 GUI software.

Parameters	Description	OBJ	AOR	ID	PRO
AND	Calculates the logical AND of the Controls connected as inputs.	-	☒	☒	☒
OR	Calculates the logical OR of the Controls connected as inputs.	-	☒	☒	☒
XOR	Calculates the logical XOR of the Controls connected as inputs.	-	☒	☒	☒
NAND	Calculates the logical NAND of the Controls connected as inputs.	-	☒	☒	☒
NOR	Calculates the logical NOR of the Controls connected as inputs.	-	☒	☒	☒
XNOR	Calculates the logical XNOR of the Controls connected as inputs.	-	☒	☒	☒
NOT	Calculates the logical NOT of the Controls connected as inputs.	-	☒	☒	☒
MAJORITY	Calculates the minimum number of input Controls whose result must be SUCCESS in order to make the Logical Tool result SUCCESS too.	-	☒	☒	☒

5.5.1 Logical Tool Selection and Configuration

Working with AOR Models, when you click on the “Output Setup” button, the OUTPUT SETUP PANEL will be displayed in the central area of the main window, allowing to insert and configure Logical Tools. You will see the list of the Controls currently inserted in the Inspection on the left side and the three digital Outputs on the right, both shown as rectangle boxes. Output boxes are highlighted in gray or in yellow depending on the logical status of the output, low or high.



Select the Logical Tool you want to add to the current Inspection using the pull-down menu in the OUTPUT SETUP PANEL under the “Select Logical Tool” label. Once selected, you have

to position your Logical Tool in the Logical Tool container. This is done graphically by left-clicking the mouse in the position you want to locate the Logical Tool; after doing that a gray shape representing the inserted Logical Tool will appear.

Three are the possible colours that a Logical Tool shape can have during configuration (see table below):

- **gray**: the Logical Tool gate is NOT CONNECTED, i.e. the number of currently connected inputs is not enough to logically process the Logical Tool. You have to add input to this Logical Tool. When the Logical Tool is NOT CONNECTED its Status is FAILURE.

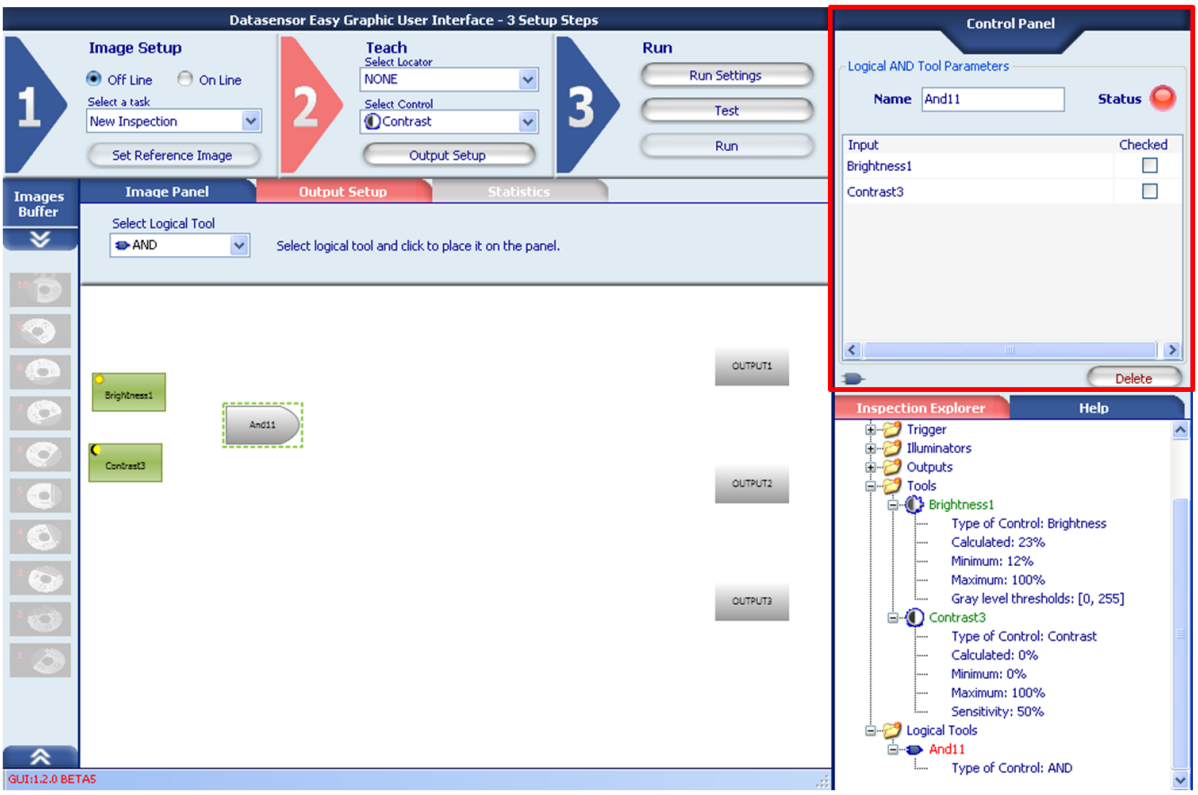
NOTE

The minimum number of inputs necessary to process each Logical Tool depends on the type of Logical Tool inserted.

- **Red**: the Logical Tool is CONNECTED and its Status is FAILURE.
- **Green**: the Logical Tool is CONNECTED and its Status is SUCCESS.

Not Connected	Connected and SUCCESS	Connected and FAILURE

Once inserted, the CONTROL PANEL will display the Logical Tool parameters you can use for the configuration:



For each Logical Tool, the CONTROL PANEL displays a table with the list of the possible Inputs that can be connected to the Logical Tool. All the Controls, the Locator and the Logical Tools currently inserted in the Inspection can be configured as Inputs for the current

selected Logical Tool. To configure Logical Tool Inputs left-click with the mouse the checkbox on the row corresponding to the Input you want to be connected to the current Logical Tool.
For each checked box, a connection line will appear in the OUTPUT SETUP PANEL between the selected inputs shape and the Logical Tool shape.
To delete the input connections, left-click with the mouse on the checked box in the corresponding row. The box then appear unchecked and the connection line deleted.

NOTE

It is not possible to create loops in the Logical Tool network; a Logical Tool A cannot be connected to a Logical Tool B that is currently configured as an Input for the Logical Tool A.

NOTE

You can always view Logical Tool parameters by selecting its symbol with the mouse from the OUTPUT SETUP PANEL.

Parameters	Description	Type	Default
Name	The Name assigned to the Logical Tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	The default name of the Logical Tool
Status	A tool result indicator . • **Green**: SUCCESS. The Logical Tool result is the Boolean TRUE. • **Red**: FAILURE. The Logical Tool result is the Boolean FALSE	Standard, Output	
Input	The list of the Controls, Locators and Logical Tools, currently inserted in the Inspection, that can be connected as inputs to the Logical Tool. ***Note***: each Logical Tool type has a minimum number of Inputs that must be connected in order to be processed. When the number of currently connected Inputs is lower than the minimum required the Status is FAILURE and the Logical tool box is gray. ***Note***: some Logical Tools have a maximum number of Inputs that can be connected and processed. When the number of connected Inputs has reach the maximum admitted the DataVS2 GUI doesn't allow you to connect inputs any more.	Standard, Input	
Checked	The list of the Controls, Locators and Logical Tools, currently inserted in the Inspection , that are already connected as inputs to the selected Logical Tool.	Standard, Input	False
Delete	Deletes the Logical Tool from the current Inspection	Standard, Input	

5.5.2 Configuration of the Image Saving Outputs

Once the Logical Tool configuration has been completed, you have to configure the outputs from the OUTPUT SETUP PANEL. Output configuration consists in connecting each of the output rectangular shapes to one of the Logical Tool or Control currently inserted in the Inspection and displayed in the OUTPUT SETUP PANEL. An output connection represented by a connection line in the OUTPUT SETUP PANEL represents the output Mode (see par. 4.3.4).

The same configuration operations can be carried out for the image saving condition (refer to par. 4.3.4).

Two are the possible colours that an output shape can have during configuration (see table below):

- **Gray**: represents the logical LOW status. The output is not connected to any Logical Tool nor Control or the output is connected and its logical value is LOW (0).
- **Yellow**: represents the logical HIGH status. The output is connected to one Logical Tool or control and its logical value is HIGH (1).

Not Connected	Connected and HIGH	Connected and LOW

To configure one of the outputs, left-click with the mouse the corresponding output's shape. Select Input and adjust other parameters from the CONTROL PANEL (par. 4.3.4).

NOTE *You do not need to configure outputs to run the Inspection on the sensor.*

NOTE *An output can be connected to only one Input.*

NOTE *In OBJ Models Output configuration can only be done through the CONTROL PANEL (see par. 4.3.4).*

5.5.3 Logical AND

Shape & Icon	Truth table	Min number of Inputs	Max number of Inputs
	INPUT / OUTPUT A B A AND B 0 0 0 0 1 0 1 0 0 1 1 1 ***Note***: in the truth table above 0 means FAILURE, 1 means SUCCESS	2	The number of the Controls, Locators and Logical Tools currently inserted in the Inspection

5.5.4 Logical OR

Shape & Icon	Truth table	Min number of Inputs	Max number of Inputs
	INPUT / OUTPUT A B A OR B 0 0 0 0 1 1 1 0 1 1 1 1 ***Note***: in the truth table above 0 means FAILURE, 1 means SUCCESS	2	The number of the Controls, Locators and Logical Tools currently inserted in the Inspection

5.5.5 Logical NOR

Shape & Icon	Truth table	Min number of Inputs	Max number of Inputs
	INPUT / OUTPUT A B A NOR B 0 0 1 0 1 0 1 0 0 1 1 0 ***Note***: in the truth table above 0 means FAILURE, 1 means SUCCESS	2	The number of the Controls, Locators and Logical Tools currently inserted in the Inspection

5.5.6 Logical XOR

Shape & Icon	Truth table	Min number of Inputs	Max number of Inputs
	INPUT / OUTPUT A | B | A XOR B 0 | 0 | 0 0 | 1 | 1 1 | 0 | 1 1 | 1 | 0 ***Note***: in the truth table above 0 means FAILURE, 1 means SUCCESS	2	2

NOTE

The Logical XOR Tool must always be connected to 2 Inputs. To change the pool of connected Inputs you can act on the table available in the CONTROL PANEL; to clear the input connection left-click with the mouse on the checked box in the corresponding line. After doing so, you will see the box unchecked and the connection line deleted.

5.5.7 Logical NAND

Shape & Icon	Truth table	Min number of Inputs	Max number of Inputs
	INPUT / OUTPUT A | B | A NAND B 0 | 0 | 1 0 | 1 | 1 1 | 0 | 1 1 | 1 | 0 ***Note***: in the truth table above 0 means FAILURE, 1 means SUCCESS	2	The number of the Controls, Locators and Logical Tools currently inserted in the Inspection

5.5.8 Logical XNOR

Shape & Icon	Truth table	Min number of Inputs	Max number of Inputs
	INPUT / OUTPUT A | B | A XNOR B 0 | 0 | 1 0 | 1 | 0 1 | 0 | 0 1 | 1 | 1 ***Note***: in the truth table above 0 means FAILURE, 1 means SUCCESS	2	2

NOTE	*The Logical XNOR Tool must always be connected to 2 Inputs. The table, available in the CONTROL PANEL, allows the user to change the pool of connected Inputs. Left-click with the mouse on the checked box in the corresponding row to eliminate the input connections. The box will then appear unchecked and the connection line deleted.*

5.5.9 Logical NOT

Shape & Icon	Truth table	Min number of Inputs	Max number of Inputs
	INPUT | OUTPUT A | NOT A 0 | 1 1 | 0 ***Note***: in the truth table above 0 means FAILURE, 1 means SUCCESS	1	1

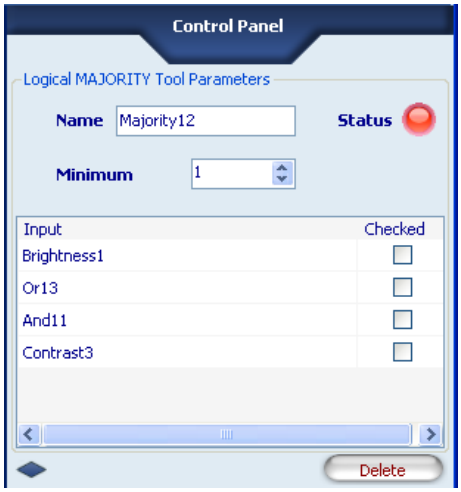
NOTE	*The table, available in the CONTROL PANEL, allows the user to change the connected Input, by just selecting a different Input checkbox.*

5.5.10 Logical MAJORITY

Majority is a logical operator acting as a comparator between the Logic result of the currently connected Inputs (that can be Locator, Controls and Logical Tools) and a numeric value representing the minimum number of Input Tools whose result that must be SUCCESS in order to make the MAJORITY result be SUCCESS.

Shape & Icon	Description	Min number of Inputs	Max number of Inputs
	If the number of the connected inputs whose result is SUCCESS is greater or equal to the "Minimum" expected value the result is SUCCESS (1) otherwise it is FAILURE (0).	1	The number of the Controls, Locators and Logical Tools currently inserted in the Inspection

The following picture and the table below display the MAJORITY Tool parameters:



Parameters	Description	Type	Default
Name	The Name assigned to the Logical Tool. The maximum length accepted is 256 UTF-8 characters. ***Note***: all the Tools currently present in the same Inspection must be identified by a univocal Name: after Name insertion, the DataVS2 GUI will check its integrity and, if the inserted name is the same as the name of another Tool currently inserted in the Inspection, it will change it by adding an integer number at the end of the Name characters.	Standard, Input	The default name of the Logical Tool
Status	A tool result indicator . • Green: SUCCESS. The Logical Tool result is the Boolean TRUE. • Red: FAILURE. The Logical Tool result is the Boolean FALSE	Standard, Output	
Input	The list of the Controls, Locators and Logical Tools, currently inserted in the Inspection, that can be connected as inputs to the Logical Tool. ***Note***: each Logical Tool type has a minimum number of Inputs that must be connected in order to be processed. When the number of currently connected Inputs is lower than the minimum required the Status is FAILURE and the Logical tool box is gray coloured. ***Note***: some Logical Tools have a maximum number of Inputs that can be connected and processed. When the number of connected Inputs has reach the maximum admitted the DataVS2 GUI doesn't allow you to connect inputs any more.	Standard, Input	
Minimum	The minimum number of connected Input whose result must be SUCCESS in order to have SUCCESS as result of the Majority Tool	Standard, Input	
Checked	The list of the Controls, Locators and Logical Tools, currently inserted in the Inspection , that are already connected as inputs to the selected Logical Tool.	Standard, Input	False
Delete	Deletes the Logical Tool from the current Inspection	Standard, Input	

6 INSPECTION SWITCHING

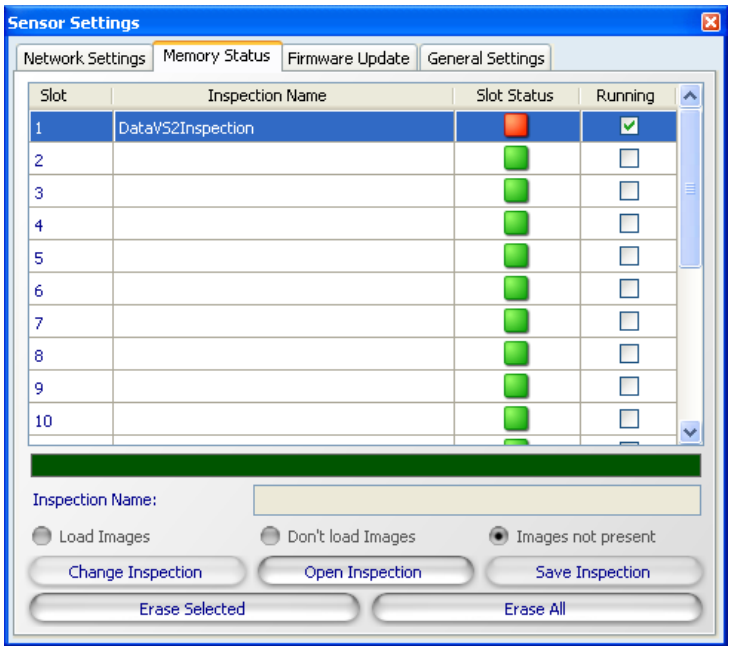
In order to change the current Inspection running on sensor you can choose between three options, described in the following paragraphs:

Inspection Change Mode	OBJ	AOR	ID	PRO
inspection switching through DataVS2 GUI	☒	☒	☒	☒
inspection switching through digital input.	☒	☒	-	-
Inspection switching through custom Ethernet communication	☒	☒	☒	☒
Switching the currently running Inspection through a command on the RS-232 serial interface.	-	-	☒	☒

The following paragraphs describe the options listed in the previous table.

6.1 INSPECTION SWITCHING THROUGH DATAVS2 GUI

When connected to the sensor, from the Menu Sensor select item Settings. The Sensor Settings dialog window will appear: select the “Memory Status” Tab.



In the top view of the window you will see a table showing the current sensor memory slots status. Each slot is represented by a table line, characterized by four columns: from left to right, the first three columns represent the Slot Number, the name of the Inspection stored in the corresponding Slot and the Slot Status, containing a red led when an Inspection is stored in the corresponding slot or a green led when the slot is empty. The last table column represent the sensor current running Inspection, corresponding to the checked row.

In order to change the current running Inspection you have to select the table row corresponding to the Inspection you want to start running and press the “Change Inspection” button, on the bottom right view of the dialog window.

Closing the window will cancel the current action; the currently running Inspection will not change.

6.2 INSPECTION SELECTION THROUGH DIGITAL INPUT

The sensor offers two different digital Inspection switching protocols:

- **Standard protocol**: inspection switching protocol without sensor acknowledgement.
- **Expert protocol**: inspection switching protocol with sensor acknowledgement. This protocol is available only in sensor firmware version higher or equal to STD.2.1.X.

NOTE

Both Standard and Expert protocols are enabled only when DataVS2 GUI is not connected to the sensor.

NOTE

All digital wave forms described for Standard and Expert protocols are intended to have a minimum duration MIN_PULS_DUR that can be configured through the DataVS2 GUI software.

6.2.1 Standard Protocol

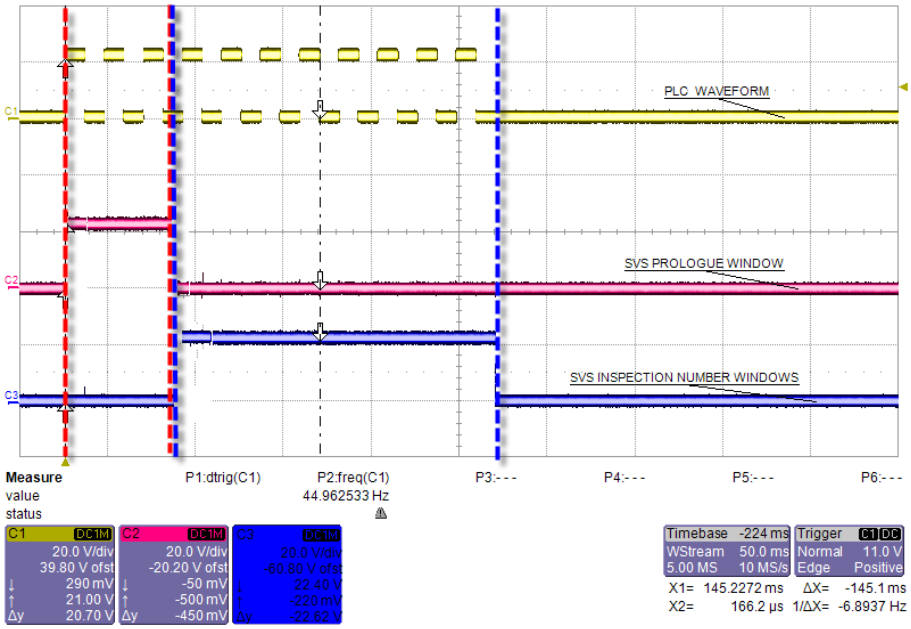
Protocol Description

The protocol is composed by a two-phases handshake, called *prologue* and *epilogue*. According to this protocol the sensor doesn't make available any ACKNOWLEDGE signal.

The PIN of the digital I/O involved in the Standard Protocol is PIN 1: Select Input. It is used in the prologue to alert the sensor of an incoming Inspection switching request and to communicate to the sensor the Slot number of the Inspection to switch, during selection.

Basically the prologue is composed of 3 pulses with a width of 10-100 ms and a duty cycle of 50 %. The first pulse starts the prologue window that lasts a max of *Y=X*6 + delta* ms (where X is the selected pulse width and *delta* is a 5% margin based on the current value of X*6), during the prologue window 3 pulses are expected to start the inspection selection window. When the prologue window is ended, if 3 and only 3 pulses were counted we proceed to the next step.

A *Y=X*40 + delta* ms time window is started and during this period pulses are counted (where X is the selected pulse width and *delta* is a 5% margin based on the current value of X*40). At the end of the period, the counted pulses mean the selected inspection (if the number fits in the available inspections slots, otherwise the request is discarded).
The following image shows an oscilloscope diagram representing the selection of the inspection 8 by the digital input generated by a PLC.

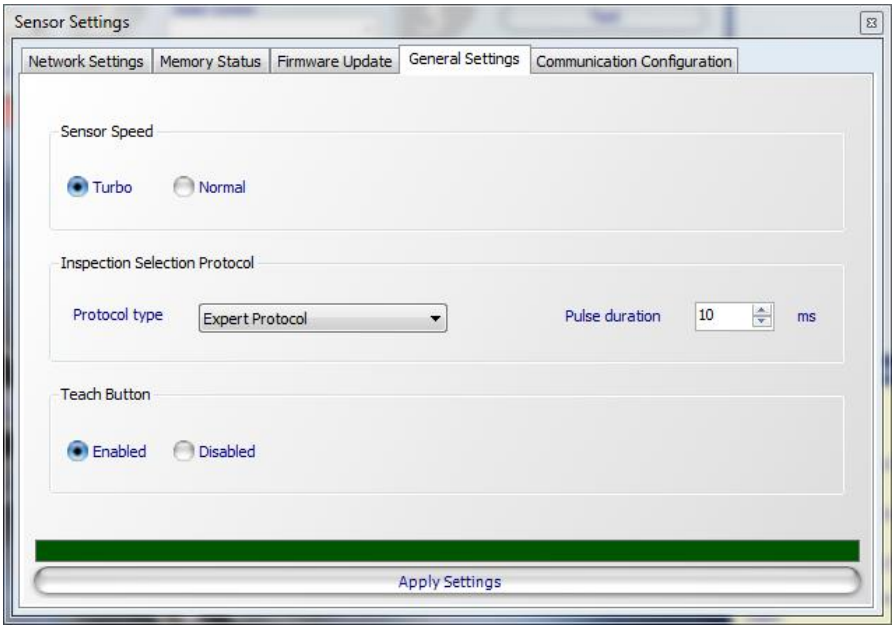


Selection of the inspection number 8

Protocol Selection

To enable the standard protocol on the sensor open the DataVS2 GUI software and connect to the sensor. Select the menu Sensor →Settings; the Sensor Settings window will be opened. Select the “General Settings” Tab.
In the “Inspection Selection Protocol” group box select “Standard protocol”, if not yet selected. Pressing the “Apply Settings” button the new configuration will be sent to the sensor and a confirmation popup message will be shown.

Use the numeric up-down control to change the minimum pulses width (MIN_PULSE_DUR) accepted by the sensor during Inspection selection through digital input handshake
Once you are confident your configuration is completed, press the “Apply Settings” button to save the configuration on the sensor. If you close the window without pressing the “Apply Settings” button, nothing will change.



Inspection Switch Timing

Because Inspection loading can take some processing time (basically the sensor needs to perform an Inspection teaching phase on the reference image before going in the run mode), a unique and definitive timing of the inspection change process is not available as it depends on the inspection time itself.

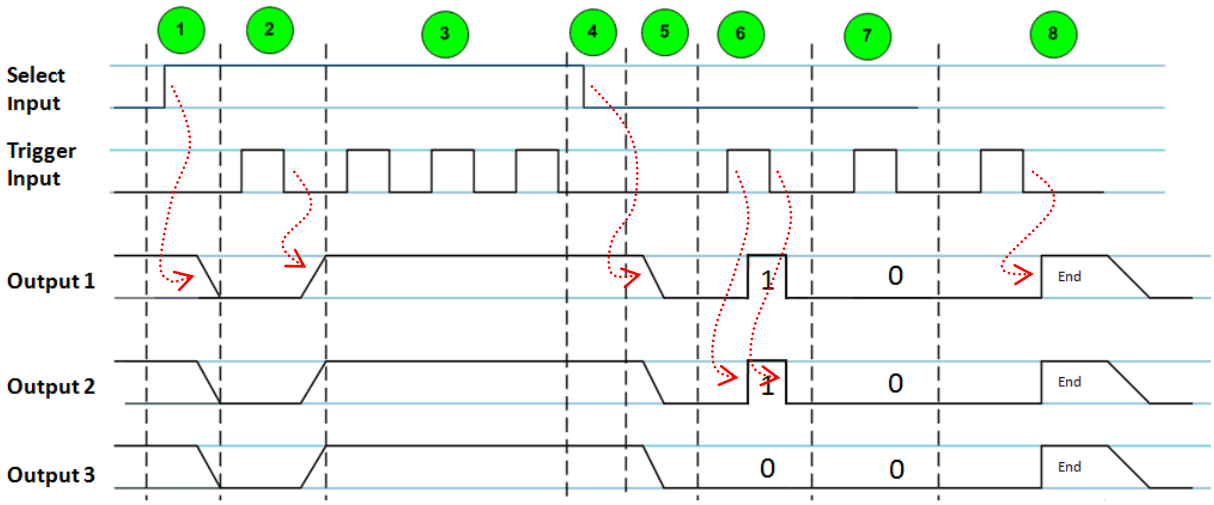
If the inspection uses very fast tools, for example Position or Width, it will be quite fast (hundreds of milliseconds), if the inspection is rather slow, for example a difficult pattern match, the delay maybe in a 1-2 seconds range.

During the inspection switch process, the sensor stops acquiring new images and enter an idle status. When the switch is complete, the sensor blinks once and then starts the new inspection activity.

The currently running inspection can be checked using the Graphic User Interface.

6.2.2 Expert Protocol

Protocol Description



Selection of Inspection number 3 (000011)

The Expert protocol for Inspection switching on the sensor is based on 8 handshaking phases. According to this protocol, the sensor generates an ACKNOWLEDGE message, providing feedback on the Inspection switching operation.

Five digital I/O PINs are involved in the Expert protocol:

- **PIN 1: Select Input**. It is used in the first 4 phases of the handshake, to start Inspection switching and set the number of the new Inspection.
- **PIN 8: Trigger Input**. It is used to communicate to the sensor the Slot number of the Inspection to switch to and to verify the correct Inspection switching ACKNOWLEDGE.
- **PIN 4: Output 1**. Together with the other Output signals, constitutes sensor feedback (responses) to the Select Input and Trigger Input signals.
- **PIN 5: Output 2**. Like Output 1.
- **PIN 6: Output 3**. Like Output 1.

The protocol requires that in each phase in which a sensor Input is changed, no further action is taken until the sensor responds with the resulting digital outputs.

A description of each phase is given below:

- **Phase 1: Select Input HIGH. ALL sensor Outputs LOW.** Starts the Inspection switching operation. The Select Input signal (from PLC to sensor) must switch from LOW (0V) to HIGH (24V) and maintain this state until the end of phase 4. Wait and do not continue with the next phase until the sensor responds by setting all Outputs to LOW.
  If the current Inspection has the "delayed output" parameter set differently from 0, then the state of the Outputs must be checked with a delay time at least equal to the delayed output parameter setting.

NOTE

In the worst case the time interval occurring between the instant when the Select Input signal arrives at the sensor and the instant when the sensor sets outputs to LOW can vary depending on the currently running Inspection. After receiving the Select Input signal, the sensor will finish the current Inspection cycle and then will set the outputs to LOW.

- **Phase 2: Trigger Input pulse LOW→HIGH→LOW. All sensor Outputs HIGH.** The Trigger Input pulse (from the PLC) indicates to the sensor to prepare to receive the new Inspection number. The sensor responds by setting all outputs to HIGH, and waits for the Inspection selection phase.

- **Phase 3: Inspection selection.** The Trigger Input pulses represent the new Inspection number (from 1 to 20). The sensor counts these incoming pulses as long as the Select Input is HIGH.

NOTE

A timeout (T1) will end Phase 3 if the Select Input signal does not go LOW within 10 seconds from when it went HIGH (Phase 1). If this timeout expires the Inspection switching operation is interrupted, the Outputs are set LOW and the sensor will resume the currently running Inspection.

- **Phase 4: Select Input LOW. End new Inspection number selection.** The next phases will verify the correct selection of the Inspection where the sensor performs an integrity check on the number of received Trigger Input pulses.

NOTE

If the number of received Trigger Input pulses is 0, the Inspection switching operation is interrupted, the Outputs are set LOW and the sensor will resume the currently running Inspection.

- **Phase 5: All sensor Outputs LOW.** After the Select Input is set to LOW by the PLC (Phase 4), the sensor sets all Outputs LOW. This transition means that the sensor ends counting the Trigger Input pulses and is now ready to send the ACKNOWLEDGE message.

NOTE

While Select Input signal is LOW but outputs are still HIGH the sensor will not accept, count or respond to any Trigger Input pulse.

The next two phases (6 and 7), make up the sensor response ACKNOWLEDGE to the Inspection number set in phase 3: the sensor responds with the number of the Inspection it will execute at the end of the protocol.
If the number of the Inspection set in phase 3 is valid, that is, in the range 1 to 20 and the selected Slot contains a valid Inspection, this number will be communicated by its 6-bit binary representation and run at phase 8.
If instead the Inspection number is not valid, that is, greater than 20 or an empty Slot, these next two phases will respond with the 6-bit binary representation of the Inspection number running prior to the start of this protocol, which will in any case be run at phase 8.
In the first part (phase 6) the first three least significant bits (LSB) are communicated, and in the second part (phase 7) the second three most significant bits (MSB) are communicated. For each of these two phases, Output 1 represents the LSB of the three bits and Output 3 represents the MSB of the three bits.

Example:

- The binary coded Inspection number is represented by 6 bits: **b5 b4 b3 b2 b1 b0**.
- In phase 6 only the three LSB bits are sent::
  Output1=b0, **Output2=b1**, **Output3=b2**
- In phase 7 only the three MSB bits are sent::
  Output1=b3, **Output2=b4**, **Output3=b5**

- **Phase 6: Trigger Input HIGH. Sensor LSB ACKNOWLEDGE BITS.** The first transition of the Trigger Input from LOW to HIGH makes the sensor respond with the three Outputs representing the three LSBs of the Inspection number. When the Trigger Input transits from HIGH to LOW, the sensor sets the three Outputs LOW.

NOTE

A timeout (T2) will end the operation if no Trigger Input signal is received within 10 seconds from the start of Phase 5. If this timeout expires the Inspection switching operation is interrupted, the Outputs are set LOW and the sensor will resume the currently running Inspection.

- **Phase 7: Trigger Input HIGH. Sensor MSB ACKNOWLEDGE BITS.** The second transition of the Trigger Input from LOW to HIGH makes the sensor respond with the three Outputs representing the three MSBs of the Inspection number. When the Trigger Input transits from HIGH to LOW, the sensor sets the three Outputs LOW.

NOTE *A timeout (T2) will end the operation if no Trigger Input signal is received within 10 seconds from the start of Phase 5. If this timeout expires the Inspection switching operation is interrupted, the Outputs are set LOW and the sensor will resume the currently running Inspection.*

- **Phase 8: Trigger Input pulse LOW→HIGH→LOW: Inspection switching execution. All sensor Outputs HIGH.** In this phase the Trigger Input pulse represents the execution of the Inspection switching. The sensor responds by setting all Outputs HIGH which will remain in this state until the end of the process. The transition of all sensor outputs from HIGH to LOW indicate to the PLC the end of the operation and that the Trigger Input (if used for this purpose) is now available to be used with the new Inspection.

NOTE *A timeout (T2) will end the operation if no Trigger Input signal is received within 10 seconds from the start of Phase 5. If this timeout expires the Inspection switching operation is interrupted, the Outputs are set LOW and the sensor will resume the currently running Inspection.*

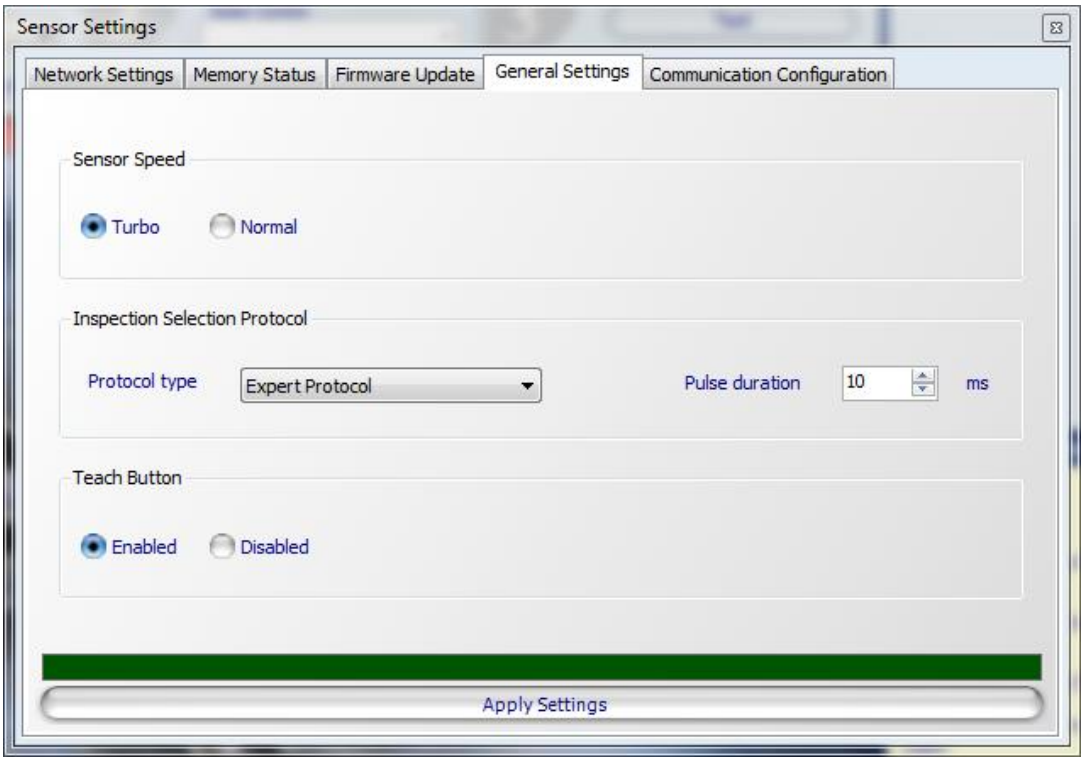
NOTE *During the actual Inspection switching while the outputs are still HIGH, the sensor will ignore any further Trigger Input pulse.*

NOTE *This phase (8) is necessary even if the Inspection number set in phase 3 is not valid (negative response in phase 6 and 7). It is necessary in order to correctly end the protocol.*

Protocol Selection

To enable the expert protocol on the sensor open the DataVS2 GUI software and connect to the sensor. Select the menu Sensor →Settings; the Sensor Settings window will be opened. Select the “General Settings” Tab.
In the “Inspection Selection Protocol” group box select “Expert protocol”, if not yet selected. Pressing the “Apply Settings” button the new configuration will be sent to the sensor and a confirmation popup message will be shown.

Use the numeric up-down control to change the minimum pulses width (MIN_PULSE_DUR) accepted by the sensor during Inspection selection through digital input handshake
Once you are certain that your configuration is completed, press the “Apply Settings” button to save the configuration on the sensor. If you close the window without pressing the “Apply Settings” button, nothing will change.



6.3 INSPECTION SWITCHING THROUGH ETHERNET COMMUNICATION

The sensor presents some utility functions to a client process connected through Ethernet communication.
In this paragraph you will find a brief description of the set of operations the client process must perform to change the current running Inspection on sensor. Please refer to Chapter 9 for further details on Ethernet communication.
To change the current running Inspection on the sensor though Ethernet communication, you have to create a PC application program performing the following operations:

- Create a socket connection between the PC and the sensor
- Send the command SETCURRENTSCRIPTID&numSlot to change the current running inspection, where numSlot is the number that indicates the slot of the inspection that you want in execution. [numSlot is a number between 0 and 19].

  ES: SETCURRENTSCRIPTID&2

- Send message DISCONNECT

The following example code represents the set of C# code instructions used to change the current running Inspection on sensor:

```
//Sensor current IPAddress
private IPAddress _sensorIP = IPAddress.Parse("172.27.101.208");
//Sensor IPPort
private int _sensorPort = 5423;
//connection socket
private TcpClient _socketClient = null;
```

```
//socket network stream
private NetworkStream _socketStream = null;
//output stream
private StreamWriter _streamWriter = null;
//input stream
private StreamReader _streamReader = null;
//create the socket connection to the sensor
_socketClient = new TcpClient();
_socketClient.NoDelay = true;
_socketClient.Connect(_sensorIP, _sensorPort);
//create the stream to read/write messages to the sensor
_socketStream = _socketClient.GetStream();
_streamReader = new StreamReader(_socketStream);
_streamWriter = new StreamWriter(_socketStream);
//set current running Inspection to 2
_streamWriter.Write("SETCURRENTSCRIPTID&2\0");
_streamWriter.Flush();
//read sensor response
_string answer = "";
int B = _socketStream.ReadByte();
while (B > 0)
{
    _answer += Convert.ToChar(B);
    B = _socketStream.ReadByte();
}
Console.WriteLine(_answer);
//disconnect from the sensor
_streamWriter.Write("DISCONNECT\0");
_streamWriter.Flush();
```

6.4 INSPECTION SWITCHING THOUGH RS232 SERIAL COMMUNICATION INTERFACE

To switch the currently running inspection in the sensor through the RS232 serial communication, you have to simply send the “SETCURRENTSCRIPTID&numSlot\0” ASCII string to the PIN RS232 Rx of DataVS2, where numSlot is the number that indicates the slot of the inspection that you want in execution. [numSlot is a number between 0 and 19].
ES: SETCURRENTSCRIPTID&2

After receiving the message, the sensor sends a reply to the RS232 Tx output.

Please refer to par. 9.3 of this manual for a detailed description of the communication protocol grammar of DataVS2.

NOTE

The switching Inspection change through RS232 interface is available only on DataVS2 ID and PRO model.

7 TEACH IN

The Teach In functionality allows to acquire a new reference image for the current inspection with no need to use the Graphical User Interface (GUI). This procedure allows to update the inspection reference image directly on the fly.

This functionality is based on two different phases:

1) Teach In activation
2) Teach In execution

Please note that a new reference image is actually acquired only when both phases are carried out properly. The Teach In execution phase (2) follows the currently selected image acquisition mode. For instance, if the current inspection is using the external rising edge trigger mode, the new reference image will be acquired only when a trigger is sent to the sensor.

For what concerns the Teach In activation phase (1), several options (listed below) are available:

Teach In mode	OBJ	AOR	ID	PRO
Mechanical Teach In	☒	☒	☒	☒
Digital Input Teach In	☒	☒	-	-
Software Teach In (Ethernet, Serial)	☒	☒	☒	☒
Teach In with VSM	☒	☒	☒	☒

The user can verify the correct execution of the first phase by checking the network LED status: when the sensor has accomplished phase 1 and is ready to acquire a new reference image, the network LED blinks until the Teach In execution is done or until the timeout occurs (60 seconds).

NOTE *The Teach In function must be enabled using DataVS2 GUI.*

7.1 MECHANICAL TEACH IN

The button enabling the Teach In operation is located on the top of the sensor body, as shown in the picture below:

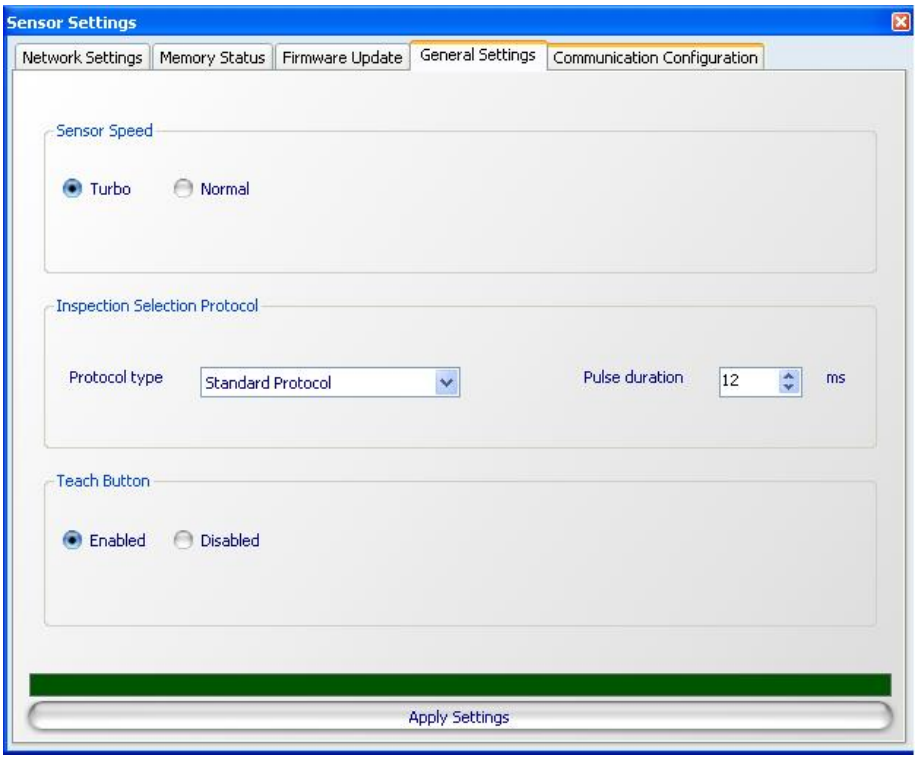
To activate the Teach In follow the steps below:

1. Shortly press the teach button.
   - the sensor stop its process. The Output 1 LED lights on as well.
   - the sensor waits for a second button pressure.

2. Press the teach button for more than one second.
   - the Output 1 LED lights off and the Outputs 2 LED lights on.
   - the connection LED is blinking until the acquisition of a new image
   - after few seconds the sensor starts again its activity.

NOTE

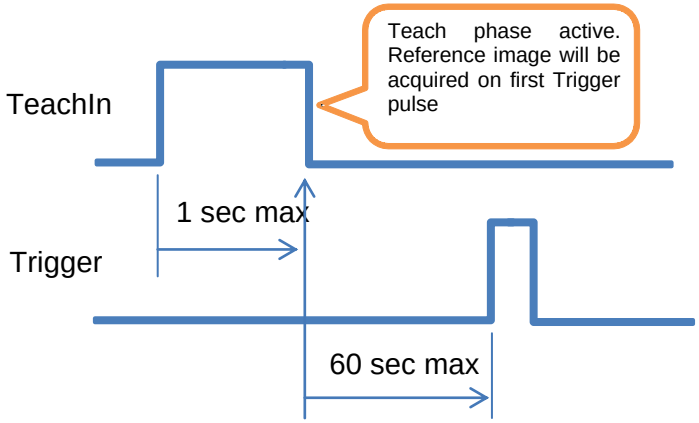
After the step 1, if the button is not pressed again or is pressed for less than 1 second the sensor returns in Run mode and starts again its activity.



To enable mechanical Teach In function through the DataVS2 GUI, from the Sensor →Settings Menu, open the "General Settings" Tab and in "Teach Button" select the "Enabled" option.

7.2 DIGITAL INPUT TEACH IN

Digital Teach In activation consists in sending to the sensor a digital pulse on PIN 1. The digital pulse must have a minimum duration specified through the DataVS2 GUI and a maximum duration of 1 sec: all pulses with this duration are accepted by the sensor.
Only after receiving the Teach In request on PIN 1, the Teach In execution phase is now active.



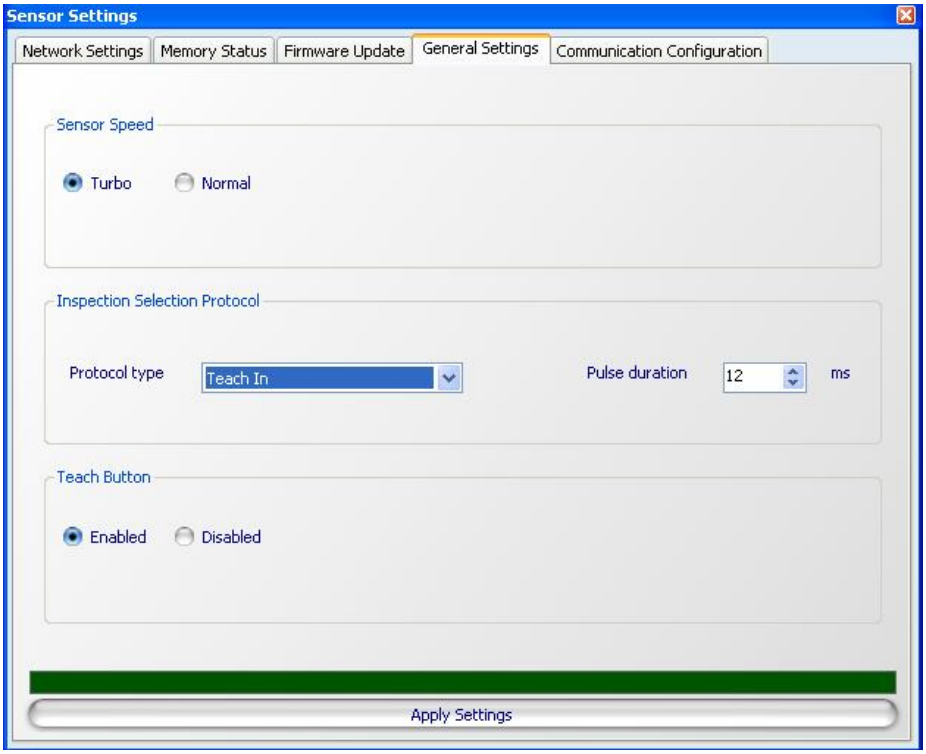
NOTE

Teach In function is available only on DataVS2 OBJ and AOR models.

NOTE

The sensor will not send any acknowledge to the digital Teach In request.

To enable digital Teach In function through the DataVS2 GUI, from the Sensor →Settings Menu, open the “General Settings” Tab and in the “Inspection Selection Protocol” group box select “Teach In” option. To set up the minimum pulse duration, please use the numeric up-down control you find in the same group box: the minimum pulse duration can vary from 10 ms to 100 ms.

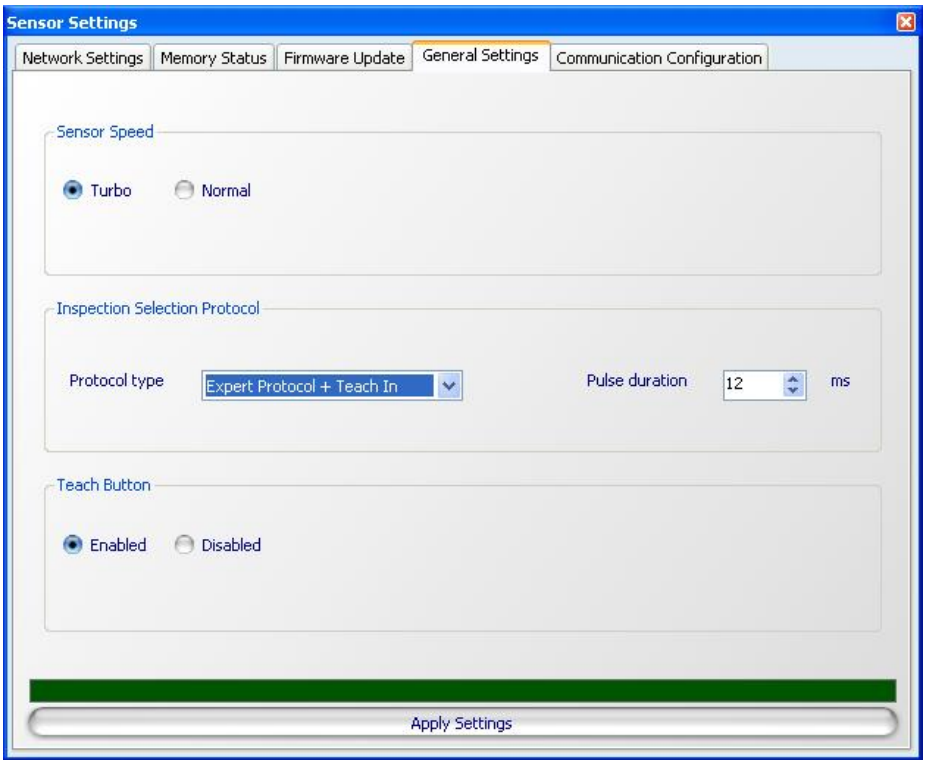


NOTE

By enabling digital Teach In function all the Inspection Switching functions will be disabled on the sensor.

If you want to keep enabled both digital Teach In and Inspection Switching functions you have to select the option “Expert protocol + Teach In” from the “General Settings” Tab of Sensor Settings window.

This option will allow you to perform the digital Teach In according to the Expert Inspection Switching Protocol (see Chapter 6): in this way the digital Teach In request corresponds to an Inspection Selection request where the slot number of the selected Inspection is the current running one.



7.3 TEACH IN SOFTWARE

To activate the software Teach In through the Ethernet connection or through the serial RS-232 connection is sufficient to send the command: “TEACH\0”. After receiving this command the sensor will enter into the execution phase.

After the acquisition of a new reference image, the sensor will send an "OK" reply message.

For a complete description of the grammar of the DataVS2communication protocol see also Chapter 9.3 of this manual.

NOTE

Software Teach In function by serial is available only on DataVS2 ID and PRO models.

The following example code represents the set of C# code instructions used to get a new reference image:

```
//Sensor current IPAddress
private IPAddress _sensorIP = IPAddress.Parse(“172.27.101.208”);
//Sensor IPPort
private int _sensorPort = 5423;
//connection socket
private TcpClient _socketClient = null;
//socket network stream
private NetworkStream _socketStream = null;
//output stream
private StreamWriter _streamWriter = null;
//input stream
private StreamReader _streamReader = null;
//create the socket connection to the sensor
_socketClient = new TcpClient();
_socketClient.NoDelay = true;
_socketClient.Connect(_sensorIP, _sensorPort);
//create the stream to read/write messages to the sensor
_socketStream = _socketClient.GetStream();
_streamReader = new StreamReader(_socketStream);
_streamWriter = new StreamWriter(_socketStream);
//set current running Inspection to 2
_streamWriter.Write("TEACH\0");
_streamWriter.Flush();
//read sensor response
_string answer = "";
int B = _socketStream.ReadByte();
while (B > 0)
{
    _answer += Convert.ToChar(B);
    B = _socketStream.ReadByte();
}
Console.WriteLine(_answer);
string[] split = _answer.Split(new char[] { '&', '\n' });
if (split[0].Equals("OK"))
{
    Console.WriteLine("TEACH send correctly.”);
}
else
{
    Console.WriteLine("Some problem in TEACH command.";
}
//disconnect from the sensor
_streamWriter.Write("DISCONNECT\0");
_streamWriter.Flush();
```

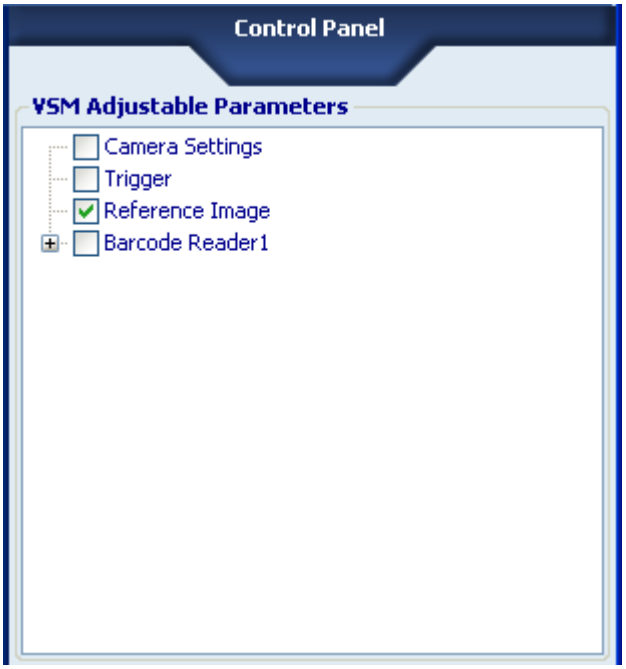
7.4 TEACH IN WITH VSM

To activate the Teach In with VSM it is necessary set up the connection to the sensor, go in Monitor mode and follow the steps below:

1. Press “Teach” Button to enter into “Adjust” mode
2. Choose “Adjust Current Inspection” option
3. From “Adjust Current Inspection” panel choose the “Reference” option
4. Press the “Teach” button

After these steps, the sensor will be ready for the acquisition of a new reference image.

To enable Teach In function with VSM through the DataVS2 GUI, click the “VSM settings” button and in the Control Panel enable the “Reference Image” option.



8 READING INSPECTION RESULTS

The following table list the options available to recover the results of the currently running inspection on the DataVS2 sensor.

Result reading mode	OBJ	AOR	ID	PRO
Reading of currently running Inspection results through DataVS2 GUI	☒	☒	☒	☒
Reading of currently running Inspection results through Ethernet interface	-	☒	☒	☒
Reading of currently running Inspection results through RS-232 serial interface	-	-	☒	☒

8.1 INSPECTION MONITORING THROUGH THE DATAVS2 GUI

Please following these instructions to monitor the current running Inspection on your sensor:

- Open the DataVS2 GUI
- from STEP 1 choose Online mode and connect to the sensor
- from STEP 1 select the Task “Open Inspection from sensor”
- select STEP 3
- From the CONTROL PANEL choose “Graphic enabled” to view results and images processed by the sensor; if you are not interested to view images choose “Statistics and timings”: the STATISTICS PANEL will be opened.
- From the CONTROL PANEL press the “Start” button: the sensor will start execution Inspection and results will be displayed by the DataVS2 GUI.

NOTE

The “Graphic enabled” option many not display in the DataVS2 GUI all results; it depends on the current PC performance and on the Ethernet connection speed. Even if you are not able to see all the running Inspection results, the sensor will run at the fastest rate possible.

8.2 READING INSPECTION RESULTS THROUGH CUSTOM ETHERNET COMMUNICATION

The sensor presents some utility functions to a client process connected through Ethernet communication. In this paragraph you will find a brief description of the set of operation the client process must perform to change the current running Inspection on sensor. Please refer to Chapter 9 for further details on Ethernet communication.
To read current running Inspection results from the sensor though Ethernet communication, you have to create a PC application program performing the following operations:

- Create a socket connection between the PC and the sensor
- Set the current sensor running status to RUN, through the command SETSTATUS
- Send the command GETSTRINGRESULTS to read the current running Inspection results.

NOTE

The sensor will reply with an UTF-8 coded string. You will need to parse it according to the protocol grammar described in the following chapter.

- Send message DISCONNECT

The following example code represents the set of C# code instructions used to read the current running Inspection results from sensor:

```
//Sensor currenty IPAddress
private IPAddress _sensorIP = IPAddress.Parse("172.27.101.208");
//Sensor IPPort
private int _sensorPort = 5423;
//connection socket
private TcpClient _socketClient = null;

//socket network stream
private NetworkStream _socketStream = null;
//output stream
private StreamWriter _streamWriter = null;
//input stream
private StreamReader _streamReader = null;
//create the socket connection to the sensor
_socketClient = new TcpClient();
_socketClient.NoDelay = true;
_socketClient.Connect(_sensorIP, _sensorPort);
//create the stream to read/write messages to the sensor
_socketStream = _socketClient.GetStream();
_streamReader = new StreamReader(_socketStream);
_streamWriter = new StreamWriter(_socketStream);
//set RUN status
_streamWriter.Write("SETSTATUS&RUN\0");
_streamWriter.Flush();
//read sensor response
_string answer = "";
int B = _socketStream.ReadByte();
while (B > 0)
{
    _answer += Convert.ToChar(B);
    B = _socketStream.ReadByte();
}
Console.WriteLine(_answer);
//send message GETSTRINGRESULTS
_streamWriter.Write("GETSTRINGRESULTS\0");
_streamWriter.Flush();
//read results
string _serverAnswer = "";
while (_serverAnswer.Equals(""))
{
   byte[] bufferByteArray = new byte[255*64];
   int nByte = _socketStream.Read(bufferByteArray, 0, 255 * 64);
   Encoding enc = Encoding.UTF8;
   _serverAnswer = enc.GetString(bufferByteArray, 0, nByte - 1);
}
Console.WriteLine(_serverAnswer);
//disconnect from the sensor
_streamWriter.Write("DISCONNECT\0");
_streamWriter.Flush();
```

8.3 READING INSPECTION RESULTS THROUGH RS232 SERIAL INTERFACE

The transmission of the results through the RS232 serial interface can be performed in two ways, that can be selected through the DataVS2 GUI interface:

- **Synchronous** transmission: in this mode the sensor sends the results of the currently running inspection only after receiving the request of the GETSTRINGRESULTS command. The results sent after the command are relative to the last execution cycle completed by the sensor.
- **Asynchronous** transmission: in this mode the sensor transmits the results of the currently running inspection at the end of each execution cycle as soon as they are available without waiting for the connected client's request.

In both cases, the results sent have to be interpreted according to the grammar described in paragraph 9.3.

9 COMMUNICATION PROTOCOLS

The following table describes the interfaces and communication protocols available in the DataVS2 models.

Interface	Description	OBJ	AOR	ID	PRO
Ethernet	The communication protocol available on the Ethernet channel is a **Synchronous** Client-Server protocol; the sensor implements a server functionality and accepts a TCP socket connection on the 5423 port. During communication, the Client sends a command to the Server according to the syntax described in paragraph 9.3; the Server (sensor) parses the command received, carries out the action and sends a Success response to the Client. If the Server is not able to parse the received command or, if the request requires an action that cannot be activated, the Server will send a Failure notice as described in the next paragraph.	☒	☒	☒	☒
Serial RS-232	The communication protocol available on the serial interface is a **Synchronous** or **Asynchronous** Client-Server protocol. The sensor implements a server functionality and awaits the commands on the RS232 Rx port. In the Synchronous mode, the Client sends a command to the Server according to the syntax described in paragraph 9.3; the Server (sensor) parses the command received, carries out the action and sends a Success response to the Client. If the Server is not able to parse the received command or, if the request requires an action that cannot be activated, the Server will send a Failure notice as described in the next paragraph. In the Asynchronous mode, the sensor sends to the RS232 Tx port the results of the currently running inspection without waiting for the connected client's requests. The format of the inspection results is described in paragraph 9.3.	-	-	☒	☒

The following table describes the sensor functions available on the Ethernet and RS232 serial interfaces in the Synchronous mode.

Command	Description	Ethernet	RS232 Serial
RUALIVE	Sends an Ethernet watchdog reset command to the sensor (the sensor resets the Watch Dog Timer and replies). The client connection is not required to maintain a steady message flow to the camera. To close the connection on the server side (basically to check the connection status) a special handshake protocol is used to check the functionality of the client software. To do this the client software has to send periodically (every 4 sec) a keep alive message to inform the sensor that it is still working. ***Note***: you must use this command in every communication session with the sensor, otherwise the communication channel will be closed by the sensor 30 seconds after Timeout. This message (like every other message) resets a special watchdog timer used on the sensor to monitor the connection status.	☒	-

Command	Description	Ethernet	RS232 Serial
GETSTATUS	Returns the current sensor status.	☒	☒
SETSTATUS	Changes the current sensor status.	☒	-
GETCURRENTSCRIPTID	Returns the memory slot number of the currently running Inspection [number between 0 and 19]	☒	☒
SETCURRENTSCRIPTID	Changes the current running Inspection.	☒	☒
GETSTRINGRESULTS	Returns the Inspection execution result related to the last processed image. Not available for OBJ model.	☒	☒
GETREADYBUSYSTATUS	Returns the execution status of the device. The device is “Busy” if it is processing an image, it is “Ready” if it has completed the execution and it is waiting for a new trigger.	☒	☒
RUREADY	Returns a result only when the device is “Ready”	☒	☒
DISCONNECT	Closes the communication with the sensor. You must always send this command at the end of the working session, before closing the socket connection.	☒	-
TRIGGER	Represents the Ethernet / Serial Trigger command; it triggers the acquisition and inspection of a new image according to the parameters set in the currently running inspection. ***Note***: the command is accepted by the sensor only when the Trigger mode, configured for the currently running inspection is of an Ethernet / Serial type (par. 4.2.6). ***Note***: the user can set a string different from the default one as the Trigger command using the Serial and Ethernet Communication Parameter panel of DataVS2 GUI (par. 3.4.6).	☒	☒
TEACH	Triggers the acquisition of a new reference image and the Teach of the current inspection with respect to the new acquired reference.	☒	☒

9.1 ETHERNET COMMUNICATION CONFIGURATION

The communication protocol available on the Ethernet channel of DataVS2 is a Synchronous Client-Server protocol; communication between the sensor and the surrounding devices is implemented after receiving a connected client’s request.

The Ethernet communication parameters have an IP address and a Subnet mask. These parameters can be configured through the DataVS2 GUI interface as described in paragraph 3.2.

To communicate with the sensor through the Ethernet interface and without using the DataVS2 GUI configuration interface, a PC application has to be created that contains the following operations:

- Creation of a TCP socket connection between PC and sensor on the 5423 port: a communication channel with the sensor has to be opened before sending any message to the sensor
- Sending of messages to the sensor according to the functionalities to implement. The sensor functionalities available on the Ethernet interface have been described in the previous paragraph.
- Sending of the DISCONNECT message at the end of the communication between PC and sensor.

The following example code represents the set of C# code instructions used to read the Slot value corresponding to the currently running inspection:

```
//Sensor current IPAddress
private IPAddress _sensorIP = IPAddress.Parse("172.27.101.208");
//Sensor IPPort
private int _sensorPort = 5423;
//connection socket
private TcpClient _socketClient = null;

//socket network stream
private NetworkStream _socketStream = null;
//output stream
private StreamWriter _streamWriter = null;
//input stream
private StreamReader _streamReader = null;

//create the socket connection to the sensor
_socketClient = new TcpClient();
_socketClient.NoDelay = true;
_socketClient.Connect(_sensorIP, _sensorPort);

//create the stream to read/write messages to the sensor
_socketStream = _socketClient.GetStream();
_streamReader = new StreamReader(_socketStream);
_streamWriter = new StreamWriter(_socketStream);

//send message GETCURRENTSCRIPTID
_streamWriter.Write("GETCURRENTSCRIPTID\0");
_streamWriter.Flush();

//read server answer
string answer = _streamReader.ReadLine();

//disconnect from the sensor
_streamWriter.Write("DISCONNECT\0");
_streamWriter.Flush();
```

9.2 RS232 SERIAL COMMUNICATION CONFIGURATION

The serial communication protocol available on the DataVS2 serial interface is a **Synchronous** or **Asynchronous** Client-Server protocol.

In the Synchronous mode the transmission of data from the sensor to the surrounding ambient is implemented only after the receipt of the connected client's request.
In the Asynchronous mode, the sensor sends the results of the currently running inspection without awaiting the request; external requests will be processed by the sensor according to the protocol described in paragraph 9.3.

The RS232 serial communication parameters including Synchronous or Asynchronous data transmission mode can be configured in the Sensor Settings panel of DataVS2 GUI as described in paragraph 3.4.6.

9.3 COMMUNICATION PROTOCOL GRAMMAR

In this paragraph we will expose the grammar that manages the protocol syntax. For this purpose we will distinguish between three types of message format:

- **Command message format**: this format is used to represent the syntax of all the request messages coming from clients.
- **Response message format**: this format is used to describe the syntax of the ASCII replies of the sensor to requests coming from client.
- **Inspection results message format**: this format is used to describe the syntax of the replies of the sensor to Inspection results requests coming from clients; this kind of messages are UTF-8 coded messages that must be parsed according to a specific grammar.

NOTE

The command and response messages are written in a BNF (Backus and Naur form) syntax.

The different meanings are given below:

- **::=** means “is defined as”
- **|** means logic “or”
- **< >** used to enclose the name of a macro category or a *non-terminal* symbol
- **[ ]** used to enclose optional terms
- **{ }** used to enclose repeated terms
- **“”** used to enclose *terminal* symbols

9.3.1 Command Message Format

The following table displays the general syntax of the command messages:

```
<Command message> ::= <command><epilogue>

<command> ::= <CommandWithParameters> | <CommandWithoutParameters>

<CommandwithParameters> ::=
<idCommandWithParameters><ampersand><parameter>

<CommandWithoutParamters> ::= <idCommandWithoutParameters>

<idCommandWithParameters> ::= “SETCURRENTSCRIPTID” | “SETSTATUS”

<idCommandWithoutParameters> ::= “GETSTRINGRESULTS” |
“GETCURRENTSCRIPTID” |  “RUALIVE” | “GETSTATUS” | “DISCONNECT” |
“GETREADYBUSYSTATUS” | “RUREADY”

<ampersand> ::= “&” (0x26 in hexadecimal)

<parameter> ::= <character>[<character>] (ASCII string representing
the parameter)

<character> ::= “a” | .. | “z” | “A” | .. |”Z” | “0” | .. | ”9”

<epilogue> ::= <NULL>

<NULL> ::= “\0” (0x00 in hexadecimal)
```

9.3.2 Response Message Format

The sensor replies positively or negatively to each command, indicating if it can activate or not the requests. An error notice is sent in presence of a negative response. The response format varies according to the command transmitted. In all cases, the response is a string of ASCII characters and foresees a response that begins with a prologue. If the response requires a return value, this value is transmitted in the same format of the Command Message, i.e. a response that will contain the error ID and an "&" as a separator followed by result parameters or error code.

The following table displays the general syntax of the Response Messages:

```
<response message> ::= <serial response> | <ethernet response>

<serial response> ::= <serial_header><type><serial_epilogue>

<ethernet response> ::= <type><epilogue>

<type> ::= <success> | <failure>

<success> ::= <SuccessID>[<ampersand><SuccessParameter>]

<failure> ::= <FailureID><ampersand><FailureParameter>

<SuccessID> ::= "OK"

<FailureID> ::= "ERR"

<ampersand> ::= "&" (0x26 in hexadecimal)

<SuccessParameter> ::= <ASCII_string> | <svs_inspection_result>
(see next paragraph)

<FailureParameter> ::= <ErrorCode>

<ASCII_string> ::= <character>[<character>] (ASCII string
representing the parameter)

<character> ::= "a" | .. | "z" | "A" | .. |"Z" | "0" | .. | "9"

<ErrorCode > ::= "BADCOMMAND" | "INVALIDSTATUS" | "PARAMNOTVALID"

<epilogue> ::= <NULL>

<serial_header> ::= <byte>[<byte>] (8 byte ASCII coded representing
the header string)

<serial_epilogue> ::= <byte>[<byte>] (8 byte ASCII coded
representing the header string)

<NULL> ::= "\0" (string terminator)
```

Error Code	Description
BADCOMMAND	The Slave received an unknown command
INVALIDSTATUS	The Salve received a known command but it is not in the right status in order to process the command received
PARAMNOTVALID	Known command but bad parameters

9.3.3 Inspection Results Message Format

The following table displays the sensor response syntax to the GETSTRINGRESULTS command.

NOTE

The GETSTRINGRESULTS command response parameters are coded as an UTF-8 string. For each Tool belonging to the same Inspection, the parameters are returned in a well-defined order according to the grammar defined below.

```
<response message> ::= <serial response> | <ethernet response>
<serial response> ::= <serial_header><body><serial_epilogue>
<ethernet response> ::= <body><epilogue>
<body> ::= <SuccessID><separator><message length><separator>
<inspection results>
<inspection results> ::= <inspection name><separator><inspection
logic result><separator><tools result><separator><logical_tools
results>
<SuccessID> ::= "OK"
<message length> ::= <UTF-8_string> (the integer representing the
<inspection results> length)
<inspection name> ::= <UTF-8_string> (the Inspection name UTF-8
coded)
<inspection logic result> ::= <UTF-8_string> (the integer
representing the Inspection logic result UTF-8 coded: 0=SUCCESS,
1=FAILURE, 14=TIMEOUT FAILURE, 250=MEMORY FAILURE)
<tools results> ::= <tool result>[<separator> <tool result>]* (UTF-
8 string representing the list of Inspection tools results)
<tool result> ::= <tool name> <separator> <tool param result>
[<separator> <tool param result>]*
<logical_tools results> ::= <tool name> <separator> <tool logic
result>
<tool name> ::= <UTF-8_string> (the tool name UTF-8 coded)
<tool logic result> ::= <UTF-8_string> (the integer representing
the tool logic result UTF-8 coded: 1=FAILURE, 0=SUCCESS)
<tool param result> ::= <UTF-8_string> (the integer, double or
float representing the tool param result UTF-8 coded: see tables
below)
<UTF-8_string> ::= <character>[<character>]* (UTF-8 string
representing the parameter)
<character> ::= "a" | .. | "z" | "A" | .. |"Z" | "0" | .. | "9"
<separator> ::= UTF-8 character (UTF-8 coded)
<serial_header> ::= <byte>[<byte>] (8 byte ASCII coded representing
the header string)
<serial_epilogue> ::= <byte>[<byte>] (8 byte ASCII coded
representing the epilogue string)
<epilogue> ::= <NULL>
<NULL> ::= "\0" (string terminator)
```

The following tables describe, for each Tool, the order of the Tool's parameters in the returned UTF-8 string result.

Brightness Tool Results List

Parameters	Description	Parameter Type	Position
Logic_Result	The general execution result. Possible values are: 0: OK, value in range 1: NOT OK, value out of range 7: NOT OK, ROI outside image	Integer	1
ROI_X_Coordinate	The X Coordinate of the Tool's ROI	Integer	2
ROI_Y_Coordinate	The Y Coordinate of the Tool's ROI	Integer	3
ROI_Alpha_Coordinate	The Alpha Coordinate of the Tool's ROI.	Double	4
Brightness_Calculated_value	The Brightness tool calculated value	Integer	5

Contrast Tool Results List

Parameters	Description	Parameter Type	Position
Logic_Result	The general execution result. Possible values are: 0: OK, value in range 1: NOT OK, value out of range 7: NOT OK, ROI outside image	Integer	1
ROI_X_Coordinate	The X Coordinate of the Tool's ROI	Integer	2
ROI_Y_Coordinate	The Y Coordinate of the Tool's ROI	Integer	3
ROI_Alpha_Coordinate	The Alpha Coordinate of the Tool's ROI.	Double	4
Contrast_Calculated_value	The Contrast tool calculated value	Integer	5

Contour Match Tool Results List

Parameters	Description	Parameter Type	Position
Logic_Result	The general execution result. Possible values are: 0: OK, value in range 1: NOT OK, value out of range 7: NOT OK, ROI outside image	Integer	1
ROI_X_Coordinate	The X Coordinate of the Tool's ROI	Integer	2
ROI_Y_Coordinate	The Y Coordinate of the Tool's ROI	Integer	3
ROI_Alpha_Coordinate	The Alpha Coordinate of the Tool's ROI.	Double	4
Binarization_threshold_value	The Contour Match Tool binarization threshold	Integer	5
Contour_points_expected	The number of Expected Contour points	Integer	6
Contour_points_found	The number of Contour point found	Integer	7
ContourMatch_calculated_value	The contour calculated value (the matching score)	Integer	8

Edge Count Tool Results List

Parameters	Description	Parameter Type	Position
Logic_Result	The general execution result. Possible values are: 0: OK, value in range 1: NOT OK, value out of range 5: NOT OK, no edge found	Integer	1
ROI_X_Coordinate	The X Coordinate of the Tool's ROI	Integer	2
ROI_Y_Coordinate	The Y Coordinate of the Tool's ROI	Integer	3
ROI_Alpha_Coordinate	The Alpha Coordinate of the Tool's ROI.	Double	4
Variation_type_value	The Edge Count variation type found: 0: both 1: White to Black 2: Black to White 3: Automatic	Integer	5
EdgeCount_Calculated_value	The Edge Count calculated value: number of points found	Integer	6
[X Coordinate, Y Coordinate](*)	The coordinates of one found point	[Integer, Integer]	7

(*) for each Point found (EdgeCount_Calculated_value)

Width Tool Results List

Parameters	Description	Parameter Type	Position
Logic_Result	The general execution result. Possible values are: 0: OK, value in range 1: NOT OK, value out of range	Integer	1
ROI_X_Coordinate	The X Coordinate of the Tool's ROI	Integer	2
ROI_Y_Coordinate	The Y Coordinate of the Tool's ROI	Integer	3
ROI_Alpha_Coordinate	The Alpha Coordinate of the Tool's ROI.	Double	4
Width_Calculated_value	The Width tool calculated value	Integer	5
Variation_type_value	The Width variation type found: 0: both 1: White to Black 2: Black to White 3: Automatic	Integer	6
Nbr_point_found_value	The number of Edges found	Integer	7
[X Coordinate, Y Coordinate](*)	The Width found point coordinates	[Integer, Integer]	8

(*) for each Point found (Nbr_point_found_value)

Position Tool Results List

Parameters	Description	Parameter Type	Position
Logic_Result	The general execution result. Possible values are: 0: OK, value in range 1: NOT OK, value out of range 5: NOT OK, no edge found	Integer	1
ROI_X_Coordinate	The X Coordinate of the Tool's ROI	Integer	2
ROI_Y_Coordinate	The Y Coordinate of the Tool's ROI	Integer	3
ROI_Alpha_Coordinate	The Alpha Coordinate of the Tool's ROI.	Double	4
Variation_type_value	The Position variation type found: 0: both 1: White to Black 2: Black to White 3: Automatic	Integer	5
Position_X_Coordinate	The X Coordinate of the Edge found	Integer	6
Position_Y_Coordinate	The Y Coordinate of the Edge found	Integer	7
Position_Calculated_value	The Position tool calculated value	Integer	8

Pattern Match Tool Results List

Parameters	Description	Parameter Type	Position
Logic_Result	The general execution result. Possible values are: 0: OK, value in range 1: NOT OK, value out of range 7: NOT OK, SEARCH AREA outside image	Integer	1
SEARCH_AREA_X_Coordinate	The X Coordinate of the Tool's SEARCH AREA	Integer	2
SEARCH_AREA _Y_Coordinate	The Y Coordinate of the Tool's SEARCH AREA	Integer	3
SEARCH_AREA_Alpha_Coordinate	The Alpha Coordinate of the Tool's SEARCH AREA	Double	4
Nbr_found_target	The number of occurrences found of the target object.	Integer	5
Found_Target_X_Coordinate (*)	The X Coordinate of the found object (Top-Left corner of the found object ROI).	Integer	6
Found_Target_Y_Coordinate (*)	The Y Coordinate of the found object (Top-Left corner of the found object ROI).	Integer	7
Found_Target_Alpha_Coordinate (*)	The Alpha Coordinate of the found object.	Double	8
Found_Target_Score (*)	The match score of the found object	Integer	9

(*) for each found occurrence of the target object (Nbr_found_target)

Geometric Pattern Match Tool Results List

Parameters	Description	Parameter Type	Position
Logic_Result	The general execution result. Possible values are: 0: OK, value in range 1: NOT OK, value out of range 7: NOT OK, SEARCH AREA outside image	Integer	1
SEARCH_AREA_X_Coordinate	The X Coordinate of the Tool's SEARCH AREA	Integer	2
SEARCH_AREA _Y_Coordinate	The Y Coordinate of the Tool's SEARCH AREA	Integer	3
SEARCH_AREA_Alpha_Coordinate	The Alpha Coordinate of the Tool's SEARCH AREA	Double	4
Nbr_found_target	The number of occurrences found of the target object.	Integer	5
Found_Target_X_Coordinate	The X Coordinate of the found object (Top-Left corner of the found object ROI).	Integer	6
Found_target_Y_Coordinate	The Y Coordinate of the found object (Top-Left corner of the found object ROI).	Integer	7
Found_Target_Alpha_Coordinate	The Alpha Coordinate of the found object.	Double	8
Found_Target_Score	The match score of the found object	Integer	9

OCV

Parameters	Description	Parameter Type	Position
Logic_Result	The general execution result. Possible values are: 0: OK, value in range 1: NOT OK, value out of range 7: NOT OK, ROI outside image 15: NOT OK, out of memory	Integer	1
ROI_X_Coordinate	The X Coordinate of the Tool's ROI	Integer	2
ROI_Y_Coordinate	The Y Coordinate of the Tool's ROI	Integer	3
ROI_Alpha_Coordinate	The Alpha Coordinate of the Tool's ROI.	Double	4
Match_Score	The resemblance in percentage between the string found and target.	Integer	5
Binarization_threshold_value	The OCV Tool's binarization threshold.	Integer	6

Data Matrix

Parameters	Description	Parameter Type	Position
Logic_Result	The general execution result. Possible values are: 0: OK, code found 1: NOT OK, no code found 7: NOT OK, SEARCH AREA outside image 24: NOT OK, string matching failed 25: NOT OK, quality check failed 26: NOT OK, nbr code found out of range	Integer	1
ROI_X_Coordinate	The X Coordinate of the Tool's ROI	Integer	2
ROI_Y_Coordinate	The Y Coordinate of the Tool's ROI	Integer	3
ROI_Alpha_Coordinate	The Alpha Coordinate of the Tool's ROI.	Double	4
Nbr_DM_Found	The number of decoded symbols in the ROI.	Integer	5
Decoded_String (*)	The decoded symbol contents.	String	6
Overall_Quality (*)	The Overall Quality value relative to the decoded symbol.	Integer	7
Detailed Quality (*)	The values of each quality parameter: Symbol Contrast, Axial Not Uniformity, Unused Error Correction e Decode Grade.	Integer [4]	8
Nbr. Found characters (*)	Number of characters read	Integer	9

(*) for each decoded symbol (Nbr_DM_Found)

Data Matrix Locator

Parameters	Description	Parameter Type	Position
Logic_Result	The general execution result. Possible values are: 0: OK, code found 1: NOT OK, no code found 7: NOT OK, SEARCH AREA outside image 24: NOT OK, string matching failed 25: NOT OK, quality check failed	Integer	1
ROI_X_Coordinate	The X Coordinate of the Tool's ROI	Integer	2
ROI_Y_Coordinate	The Y Coordinate of the Tool's ROI	Integer	3
ROI_Alpha_Coordinate	The Alpha Coordinate of the Tool's ROI.	Double	4
Decoded_String	The decoded symbol contents.	String	5
Overall_Quality	The Overall Quality value relative to the decoded symbol.	Integer	6
Detailed Quality	The values of each quality parameter: Symbol Contrast, Axial Not Uniformity, Unused Error Correction e Decode Grade.	Integer [4]	7
Nbr. Found characters	Number of characters read	Integer	8
Found_Target_X_Coordinate	The X Coordinate of the Datamatrix symbol found (Barycenter of the found Datamatrix)	Integer	9
Found_Target_Y_Coordinate	The Y Coordinate of the Datamatrix symbol found (Barycentre of the found Datamatrix)	Integer	10
Found_Target_Alpha_Coordinate	The calculated angle of the Datamatrix symbol found	Integer	11

Barcode

Parameters	Description	Parameter Type	Position
Logic_Result	The general execution result. Possible values are: 0: OK, code found 1: NOT OK, no code found 7: NOT OK, SEARCH AREA outside image 24: NOT OK, string matching failed 25: NOT OK, quality check failed 26: NOT OK, nbr code found out of range	Integer	1
ROI_X_Coordinate	The X Coordinate of the Tool's ROI	Integer	2
ROI_Y_Coordinate	The Y Coordinate of the Tool's ROI	Integer	3
ROI_Alpha_Coordinate	The Alpha Coordinate of the Tool's ROI.	Double	4
Nbr_BC_Found	The number of decoded symbols in the ROI.	Integer	5
BC_Code_Type	The found symbology code; symbology codes are listed in the table below.	Integer	6
Decoded_String (*)	The decoded symbol contents.	String	7
Overall_Quality (*)	The Overall Quality value relative to the decoded symbol. (**)	Integer	8
Detailed Quality (*)	The values of each quality parameter: Decode, Symbol Contrast, Minimum Edge Contrast, Modulation and Decodability.	Integer [5]	9
Nbr. Found characters (*)	Number of characters read	Integer	10

(*) for each decoded symbol (Nbr_BC_Found)

(**) when the quality parameters are not calculated the Overall Quality value is set to 5. (par. 5.4.10)

Barcode Locator

Parameters	Description	Parameter Type	Position
Logic_Result	The general execution result. Possible values are: 0: OK, code found 1: NOT OK, no code found 7: NOT OK, SEARCH AREA outside image 24: NOT OK, string matching failed 25: NOT OK, quality check failed	Integer	1
ROI_X_Coordinate	The X Coordinate of the Tool's ROI	Integer	2
ROI_Y_Coordinate	The Y Coordinate of the Tool's ROI	Integer	3
ROI_Alpha_Coordinate	The Alpha Coordinate of the Tool's ROI.	Double	4
BC_Code_Type	The found symbology code; symbology codes are listed in the table below.	Integer	5
Decoded_String	The decoded symbol contents.	String	6
Overall_Quality	The Overall Quality value relative to the decoded symbol. (*)	Integer	7
Detailed Quality	The values of each quality parameter: Decode, Symbol Contrast, Minimum Edge Contrast, Modulation and Decodability.	Integer [5]	8
Nbr. Found characters	Number of characters read	Integer	9
Found_Target_X_Coordinate	The X Coordinate of the Barcode symbol found (Barycenter of the found Barcode)	Integer	10
Found_Target_Y_Coordinate	The Y Coordinate of the Barcode symbol found (Barycenter of the found Barcode)	Integer	11
Found_Target_Alpha_Coordinate	The calculated angle of the Barcode symbol found	Integer	12

(*) when the quality parameters are not calculated the Overall Quality value is set to 5. (par. 5.3.8)

Barcode Symbology Conversion Table

BC_Code_Type	Symbology
0	Code 128
1	Code 39
2	Codabar
3	Interleaved 2 of 5
4	EAN 13
5	EAN 8
6	UPC-E
7	Postnet
8	IMB
9	Pharmacode
10	PDF 417
11	UPC-A
12	GS1-128

QRCode

Parameters	Description	Parameter Type	Position
Logic Result	The general execution result. Possible values are: 0: OK, code found 1: NOT OK, no code found 7: NOT OK, SEARCH AREA outside image 24: NOT OK, string matching failed 25: NOT OK, quality check failed	Integer	1
ROI_X_Coordinate	The X Coordinate of the Tool's ROI	Integer	2
ROI_Y_Coordinate	The Y Coordinate of the Tool's ROI	Integer	3
ROI_Alpha_Coordinate	The Alpha Coordinate of the Tool's ROI.	Double	4
Nbr_BC_Found	Il numero di simboli decodificati all'interno della ROI	Integer	5
QR_Code_Type	The type of QR code searched, it could be QR or Micro QR; the code types are listed in the table below.	Integer	6
Decoded_String (*)	The decoded symbol contents.	String	7
Overall_Quality (*)	The Overall Quality value relative to the decoded symbol. (*)	Integer	8
Detailed Quality (*)	I valori di ciascun parametro di qualità: Symbol Contrast, Axial Not Uniformity, Unused Error Correction e Decode Grade.	Integer [4]	9
Nbr. Found characters (*)	Number of characters read.	Integer	10

(*) per ogni simbolo decodificato (Nbr_BC_Found)

QRCode Locator

Parameters	Description	Parameter Type	Position
Logic Result	The general execution result. Possible values are: 0: OK, code found 1: NOT OK, no code found 7: NOT OK, SEARCH AREA outside image 24: NOT OK, string matching failed 25: NOT OK, quality check failed	Integer	1
ROI_X_Coordinate	The X Coordinate of the Tool's ROI	Integer	2
ROI_Y_Coordinate	The Y Coordinate of the Tool's ROI	Integer	3

Parameters	Description	Parameter Type	Position
ROI_Alpha_Coordinate	The Alpha Coordinate of the Tool's ROI.	Double	4
QR_Code_Type	The type of QR code searched, it could be QR or Micro QR; the code types are listed in the table below.	Integer	5
Decoded_String	The decoded symbol contents.	String	6
Overall_Quality	The Overall Quality value relative to the decoded symbol. (*)	Integer	7
Detailed Quality	I valori di ciascun parametro di qualità: Symbol Contrast, Axial Not Uniformity, Unused Error Correction e Decode Grade.	Integer [4]	8
Nbr. Found characters	Number of characters read.	Integer	9
Found_Target_X_Coordinate	The X Coordinate of the QRCode symbol found (Barycenter of the found QRCode)	Integer	10
Found_target_Y_Coordinate	The Y Coordinate of the QRCode symbol found (Barycenter of the found QRCode)	Integer	11
Found_Target_Alpha_Coordinate	The calculated angle of the QRCode symbol found	Integer	12

QRCode Type Conversion Table

QR_Code_Type	Symbology
0	QR
1	Micro QR

360° Contour Match / 360° Contour Counter / 360° Defect Finder Tool Results List

Parameters	Description	Parameter Type	Position
Logic_Result	The general execution result. Possible values are: 0: OK, value in range 1: NOT OK, value out of range 7: NOT OK, SEARCH AREA outside image	Integer	1
ROI_X_Coordinate	The X Coordinate of the Tool's ROI	Integer	2
ROI_Y_Coordinate	The Y Coordinate of the Tool's ROI	Integer	3
ROI_Alpha_Coordinate	The Alpha Coordinate of the Tool's ROI.	Double	4
Nbr_found_target	The number of occurrences found of the target object.	Integer	5
Found_Target_X_Coordinate (*)	The X Coordinate of the found object (Barycenter of the found object ROI).	Integer	6
Found_target_Y_Coordinate (*)	The Y Coordinate of the found object (Barycenter of the found object ROI).	Integer	7
Found_Target_Alpha_Coordinate (*)	The Alpha Coordinate of the found object.	Double	8
Found_Target_Score (*)	The match score of the found object	Integer	9
Capture_Point_X_Coordinate(*)	X coordinate of the acquisition point of the object found	Integer	10
Capture_Point_Y_Coordinate(*)	Y coordinate of the acquisition point of the object found	Integer	11

(*) for each decoded object (Nbr_found_target)

9.3.4 Inspection Results Message Configuration Through DataVS2 GUI

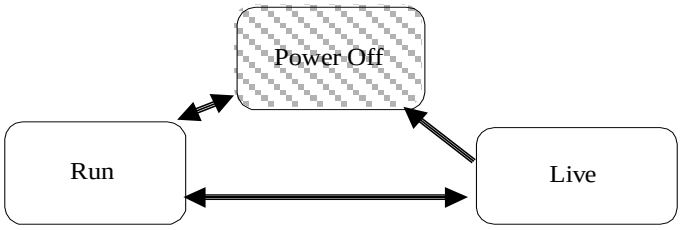
The DataVS2 GUI interface allows to configure the format of the result string returned by the sensor after receiving the GETSTRINGRESULTS command, when the Synchronous transmission mode is set or at the end of each inspection in the Asynchronous mode.

To configure the result string format you have to go into the Expert section of the Output Panel, STEP2 (par. 4.3.4), and select (check) the boxes corresponding to the parameters to be read in the result string. The result string will always contain only the parameters selected for the current inspection in the order described in the Position column in the previous tables.

9.4 COMMANDS DESCRIPTION

The sensor programming model supports the following operating status:

- **RUN**: in this status the sensor is working in the *sensor mode*, acquiring and processing images according to the currently running Inspection setting.
- **LIVE**: in this status the sensor is working in the *acquisition mode*, acquiring images on request of the client and executing commands to set up the sensor. In the LIVE STATUS the sensor does not perform any Inspection.



In the following tables you will find a description for each command available.

NOTE	*For each command the sensor STATUS represents the legal status of the sensor in order to perform the requested operation. If the sensor STATUS doesn't corresponds to the legal one, an error message will be returned as a response to the command request. To change the sensor current status, please use the SETSTATUS command.*

RUALIVE		
Parameters	*Response*	**IMALIVE**
	STATUS	**LIVE, RUN**
This command allows the user to send an ALIVE message to the sensor; the sensor resets the Watch Dog timer and sends a reply to the client. If the sensor doesn't receive any message for an interval bigger than 30 s the connection with a client will be aborted Example		
RUALIVE	IMALIVE	

DISCONNECT		
Parameters	*Response*	
	STATUS	**LIVE**
This command allows the client to disconnect from sensor. After receiving this message the sensor immediately closes the communication without any reply.		
Example		
DISCONNECT		

GETSTATUS		
Parameters	*Response*	**"OK&status"** **"ERR&BADCOMMAND"**
	status	The ASCII representation of the current working status to set on the sensor [LIVE | RUN]
	STATUS	**LIVE,RUN**
This command allows the reading of the current status of the sensor		
Example		
GETSTATUS	OK&LIVE	

SETSTATUS&status		
Parameters	*Response*	**"OK"** **"ERR&BADCOMMAND"**
status	The ASCII representation of the current working status to set on the sensor [LIVE | RUN]	
	STATUS	**LIVE,RUN**
This command allows the change the current status on the sensor		
Example		
SETSTATUS&LIVE	OK	

GETCURRENTSCRIPTID		
Parameters	*Response*	**"OK&scriptId"** **"ERR&BADCOMMAND"**
	scriptId	The ASCII string representation of the memory slot where the currently running script is stored: a value between 0 and 19.
	STATUS	**LIVE**
This command allows the user to get the value of the current script running on the sensor Example		
GETCURRENTSCRIPTID	OK&1	

SETCURRENTSCRIPTID&scriptId		
Parameters	*Response*	**"OK"** **"ERR&BADCOMMAND"**
scriptId — The ASCII string representation of the memory slot where the currently running script is stored: a value between 0 and 19.		
	STATUS	**LIVE**
This command allows the user to set the value of the current script running on the sensor Example		
SETCURRENTSCRIPTID&2	OK	

GETSTRINGRESULTS		
Parameters	*Response*	**"OK&string_res"** **"ERR&BADCOMMAND"**
	string_res	The UTF-8 string representation of the of the Inspection results.
	STATUS	**RUN**
This command allows the user to retrieve the UTF-8 coded string representing the current running Inspection results. The example below represents the sensor reply related to the current running Inspection. The current running Inspection contains a Brightness tool. Example		
GETSTRINGRESULTS	OK&"SvsInspection&00000&Brightness1&00000&00255&00300&0000.0&00180"	

GETREADYBUSYSTATUS		
Parameters	*Response*	**"OK&status"** **"ERR&BADCOMMAND"**
	status	The ASCII string representation of the current execution status of the sensor [READY | BUSY]
	STATUS	**LIVE,RUN**
This command allow the user to know if the sensor is processing an image (BUSY) or it is waiting for a new trigger (READY). Example		
GETREADYBUSYSTATUS	OK&READY	

RUREADY		
Parameters	*Response*	**"OK&READY"**
	STATUS	**LIVE,RUN**
This command returns OK&READY as soon as it can handle a new trigger. Example		
RUREADY	OK&READY	

10 RECOVERY MODE

Whenever the sensor suffers a crash due to a bad inspection definition, it is possible to load a recovery firmware with a basically communication service to restore a working environment.

To launch this recovery routine hold down the button on the top of the sensor cover during the sensor power up until the outputs LEDs start blinking, signalling that the recovery firmware is running.

The sensor can be connected through the DataVS2 GUI in the recovery condition and perform the following operations:

- overwrite the damaged Inspection by saving in the same memory Slot the recovery Inspection.
- open the damaged Inspection from the sensor and save it on the PC
- delete all the sensor Inspections (only with recovery firmware version higher or equal to STD.2.1.X.R).

NOTE

This function is always available and it is not necessary to enable it through the DataVS2 GUI to activate the Recovery Mode.

NOTE

Please enter Recovery Mode only if the sensor is not working and you are not able to establish a communication with the sensor via the DataVS2 Graphic User Interface.

NOTE

The Recovery Mode is only intended for troubleshooting. The sensor in Recovery Mode cannot be used for machine control.

NOTE

When the sensor is powered on in Recovery Mode it has the following values of IP Address and Subnet Mask: 172.27.101.208 and 255.255.0.0.

10.1 RECOVERY PROCEDURE

In this paragraph you will find a simple recovery procedure to follow in case of sensor troubles.

10.1.1 Recovery Procedure for DataVS2 Basic Sensor

Please follow the steps given below whenever your sensor is a DataVS2 OBJ model and is not working properly:

1. Power on the sensor in Recovery Mode.
2. Open the DataVS2 GUI
3. From the STEP 1 select Online and connect to the sensor
4. From the STEP 1 select the “Open Inspection from sensor” Task and load the currently running Inspection in the DataVS2 GUI
5. From the menu File→Save→Save on PC save the current Inspection on your PC.
6. From STEP 1 select “Open Inspection from PC” Task and open the Recovery Inspection (“ORRescueInspection.svs”); you will find the Recovery Inspection in the DataVS2 GUI installation folder directory on your PC.
7. Save the Recovery Inspection in the sensor: in menu File→Save→Save on sensor. Please save the Inspection in the current memory Slot.
8. Disconnect from the sensor.
9. Re-power on the sensor, in normal mode (without pressing the Recovery button).
10. Verify that the sensor is working properly: you should see both LEDs blinking and you should be able to connect to the sensor through the DataVS2 GUI.

NOTE

If you are not able to connect to the sensor through the DataVS2 GUI, please contact Sales Service.

11. Send the Inspection saved on your PC to the Datalogic Sale Service.

10.1.2 Recovery Procedure for DataVS2 Advanced Sensor

Please follow the steps given below whenever your sensor is a DataVS2 Advanced model and is not working properly:

1. Power on the sensor in Recovery Mode.
2. Open the DataVS2 GUI
3. From the STEP 1 select Online and connect to the sensor
4. From the STEP 1 select the “Open Inspection from sensor” Task and load the currently running Inspection in the DataVS2 GUI
5. From the menu File→Save→Save on PC save the current Inspection on your PC.
6. From STEP 1 select “Open Inspection from PC” Task and open the Recovery Inspection (“AORRescueInspection.svs2”); you will find the Recovery Inspection in the DataVS2 GUI installation folder directory on your PC.
7. Save the Recovery Inspection in the sensor: in menu File→Save→Save on sensor. Please save the Inspection in the current memory Slot.
8. Disconnect from the sensor.
9. Re-power on the sensor, in normal mode (without pressing the Recovery button).

10. Verify that the sensor is working properly: you should see the internal Illuminator flashes and you should be able to connect to the sensor through the DataVS2 GUI interface.

NOTE

If you are not able to connect to the sensor through the DataVS2 GUI, please contact Sales Service.

11. Send the Inspection saved on your PC to the Datalogic Sale Service.

10.1.3 Recovery Procedure for DataVS2 Identification Sensor

Please follow the following steps whenever the DataVS2 ID sensor is not working properly:

1. Power on the sensor in Recovery Mode.
2. Open the DataVS2 GUI
3. From the STEP 1 select Online and connect to the sensor
4. From the STEP 1 select the “Open Inspection from sensor” Task and load the currently running Inspection in the DataVS2 GUI
5. From the menu File→Save→Save on PC save the current Inspection on your PC.
6. From STEP 1 select “Open Inspection from PC” Task and open the Recovery Inspection (“IDRescueInspection.svs2”); you will find the Recovery Inspection in the DataVS2 GUI installation folder directory on your PC.
7. Save the Recovery Inspection in the sensor: in menu File→Save→Save on sensor. Please save the Inspection in the current memory Slot.
8. Disconnect from the sensor.
9. Re-power the sensor in normal mode (without pressing the Recovery button).
10. Verify that the sensor is working properly: you should see the internal Illuminator flashes and you should be able to connect to the sensor through the DataVS2 GUI interface.

NOTE

If you are not able to connect to the sensor through the DataVS2 GUI, please contact Sales Service.

11. Send the Inspection saved on your PC to the Datalogic Sale Service.

10.1.4 Recovery Procedure for DataVS2 Professional Sensor

Please follow the steps given below whenever the DataVS2 Professional sensor is not working properly:

1. Power on the sensor in Recovery Mode.
2. Open the DataVS2 GUI
3. From the STEP 1 select Online and connect to the sensor
4. From the STEP 1 select the “Open Inspection from sensor” Task and load the currently running Inspection in the DataVS2 GUI
5. From the menu File→Save→Save on PC save the current Inspection on your PC.
6. From STEP 1 select “Open Inspection from PC” Task and open the Recovery Inspection (“PRORescueInspection.svs2”); you will find the Recovery Inspection in the DataVS2 GUI installation folder directory on your PC.
7. Save the Recovery Inspection in the sensor: in menu File→Save→Save on sensor. Please save the Inspection in the current memory Slot.
8. Disconnect from the sensor.
9. Re-power the sensor in normal mode (without pressing the Recovery button).
10. Verify that the sensor is working properly: you should see the internal Illuminator flashes and you should be able to connect to the sensor through the DataVS2 GUI interface.

NOTE

If you are not able to connect to the sensor through the DataVS2 GUI, please contact Sales Service.

11. Send the Inspection saved on your PC to the Datalogic Sale Service.

11 INSPECTIONS AND PERIODIC MAINTENANCE

Correct system maintenance consists in removing dust or foreign objects from sensor and keeping the configuration software updated to the latest release.

Remove any dust from sensor body using a soft cloth; if necessary, dampen the cloth slightly with a mild detergent solution.

Use an anti-static compressed air can to remove dust and fingerprints from the lenses. Use a cloth and a specific lens detergent to remove any residual material.

Avoid using:
alcohol-based detergents or solvents;
wool or synthetic cloth.

Visit the web site of Datalogic (*www. datalogic.com*) to check for software updates. To check version and date of the software installed on your system, see Help → About.

11.1 WARRANTY

The warranty period for this product is 24 months.
See the General Terms and Conditions of Sales at www.datalogic.com for further details.

Warranty does not cover damage and defects resulting from improper use of the sensor or accidental damage due to shock or in the event the product is dropped.

The sensor contains cutting-edge electronic components that cannot be repaired or replaced by the final user. Do not dismantle the device or make any changes to its components or the warranty will become null and void.

Datalogic S.p.a. may not be held liable for damage to persons or property resulting from failure to comply with the provisions reported herein.

In the event of problems, contact your Datalogic representative or our Technical Service through the website.

12 SPECIFICATIONS

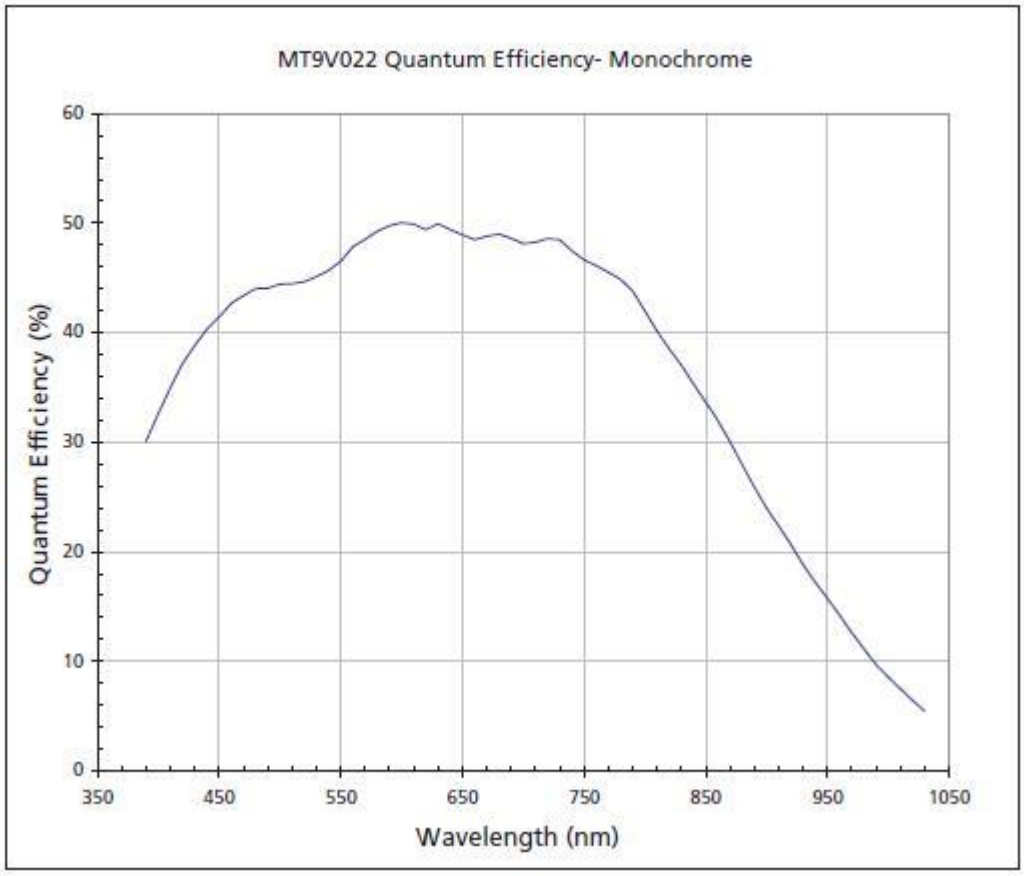
12.1 DATAVS2 SENSOR SPECIFICATION

Item	Description
Supply voltage	24 VDC ± 10%
Ripple voltage	1 Vpp max with illuminator 2 Vpp max without illuminator
Current draw *(excluding output current and illuminator)*	100 mA at 24 VDC
Current draw with illuminator *(depends on how long illuminator stays on)*	< 200 mA at 24 VDC
Outputs:	3 PNP outputs 1 PNP output that can be used as an alternative to the Strobe signal for the external Illuminator (short-circuit protection)
Network interface	M12 4-pole – 10/100 Mbps Ethernet
Interface for external illuminator	Strobe signal (24 V PNP N.O.)
Output current	100 mA max
Output saturation voltage	< 2 V
Optic	Integrated
Resolution	640 × 480 (VGA)
Indicators	4 LEDs
Setup	1 Teach-In key
Data retention	Non-volatile FLASH memory
Operating temperature	-10 °C … +55 °C
Storage temperature	-25 °C … +75 °C
Vibration (EN60068-2-6) 2 hours on each axis	14 mm @ 2 to 10 Hz 1.5 mm @ 13 to 55 Hz 2 g @ 70 to 200 Hz
Shock resistance (EN60068-2-27)	11 ms (30 G) 6 shock for every axis
Housing material	Aluminium alloy / ABS
Mechanical protection	IP50
Connection	M12 8-pole A-code, M12 4-pole D-code
Weight	125 g.

12.2 IMAGER SENSOR SPECIFICATION

Item	Description
Technology	CMOS
Supply voltage	3.3 V
Size	⅓-inch
Pixel matrix	752 x 480
Pixel size	6.0 µm x 6.0 µm
Maximum frame rate	60 fps
SNR	55 dB
Responsivity	4.8 V / lux-sec

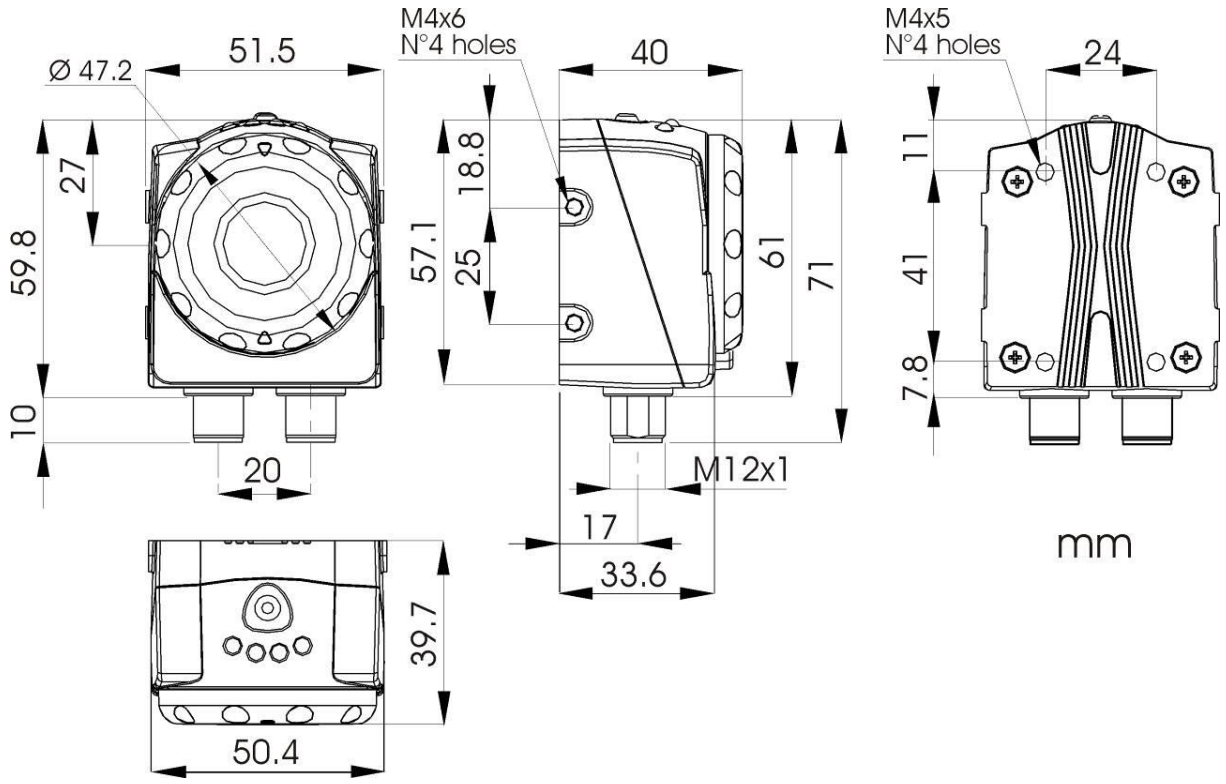
Typical Quantum Efficiency—Monochrome



13 LIST OF AVAILABLE MODELS

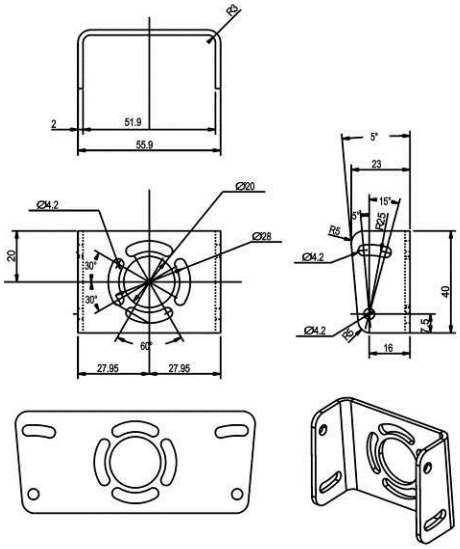
Model	Optic	Software	Output	Input	Interface	Illuminator
DataVS2-06-DE-OBJ	6 mm	Basic	3 digital output 1 strobe output	1 trigger input 1 digital input	Basic Eth.	Red
DataVS2-08-DE-OBJ	8 mm	Basic	3 digital output 1 strobe output	1 trigger input 1 digital input	Basic Eth.	Red
DataVS2-12-DE-OBJ	12 mm	Basic	3 digital output 1 strobe output	1 trigger input 1 digital input	Basic Eth.	Red
DataVS2-16-DE-OBJ	16 mm	Basic	3 digital output 1 strobe output	1 trigger input 1 digital input	Basic Eth.	Red
DataVS2-06-DE-AOR	6 mm	Advanced	3 digital output 1 strobe output	1 trigger input 1 digital input	Adv. Eth.	Red
DataVS2-08-DE-AOR	8 mm	Advanced	3 digital output 1 strobe output	1 trigger input 1 digital input	Adv. Eth.	Red
DataVS2-12-DE-AOR	12 mm	Advanced	3 digital output 1 strobe output	1 trigger input 1 digital input	Adv. Eth.	Red
DataVS2-16-DE-AOR	16 mm	Advanced	3 digital output 1 strobe output	1 trigger input 1 digital input	Adv. Eth.	Red
DataVS2-06-RE-ID	6 mm	Identification	2 digital outputs 1 strobe output	1 trigger input	RS232- Adv. Eth.	Red
DataVS2-08-RE-ID	8 mm	Identification	2 digital outputs 1 strobe output	1 trigger input	RS232- Adv. Eth.	Red
DataVS2-12-RE-ID	12 mm	Identification	2 digital outputs 1 strobe output	1 trigger input	RS232- Adv. Eth.	Red
DataVS2-16-RE-ID	16 mm	Identification	2 digital outputs 1 strobe output	1 trigger input	RS232- Adv. Eth.	Red
DataVS2-06-RE-PRO	6 mm	Professional	2 digital output 1 strobe output	1 trigger input	RS232- Adv. Eth.	Red
DataVS2-08-RE-PRO	8 mm	Professional	2 digital output 1 strobe output	1 trigger input	RS232- Adv. Eth.	Red
DataVS2-12-RE-PRO	12 mm	Professional	2 digital output 1 strobe output	1 trigger input	RS232- Adv. Eth.	Red
DataVS2-16-RE-PRO	16 mm	Professional	2 digital output 1 strobe output	1 trigger input	RS232- Adv. Eth.	Red
DataVS2-06-DE-AOR-I	6 mm	Advanced	3 digital output 1 strobe output	1 trigger input 1 digital input	Adv. Eth.	Infrared
DataVS2-08-DE-AOR-I	8 mm	Advanced	3 digital output 1 strobe output	1 trigger input 1 digital input	Adv. Eth.	Infrared
DataVS2-12-DE-AOR-I	12 mm	Advanced	3 digital output 1 strobe output	1 trigger input 1 digital input	Adv. Eth.	Infrared
DataVS2-16-DE-AOR-I	16 mm	Advanced	3 digital output 1 strobe output	1 trigger input 1 digital input	Adv. Eth.	Infrared
DataVS2-06-RE-PRO-I	6 mm	Professional	2 digital output 1 strobe output	1 trigger input	RS232- Adv. Eth.	Infrared
DataVS2-08-RE-PRO-I	8 mm	Professional	2 digital output 1 strobe output	1 trigger input	RS232- Adv. Eth.	Infrared
DataVS2-12-RE-PRO-I	12 mm	Professional	2 digital output 1 strobe output	1 trigger input	RS232- Adv. Eth.	Infrared
DataVS2-16-RE-PRO-I	16 mm	Professional	2 digital output 1 strobe output	1 trigger input	RS232- Adv. Eth.	Infrared

14 OVERALL DIMENSIONS

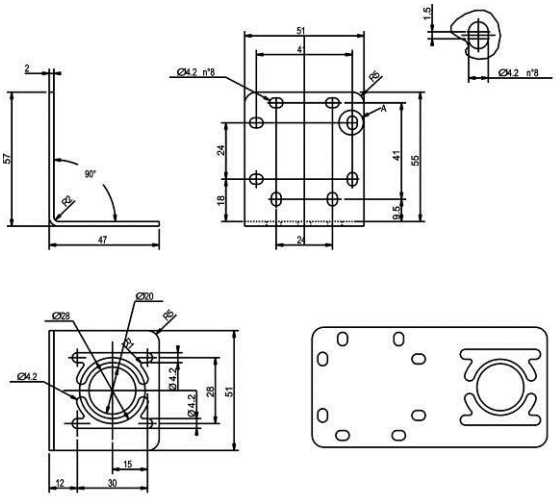


15 ACCESSORIES

ST-5066
U-shaped fixing bracket for angle adjustment

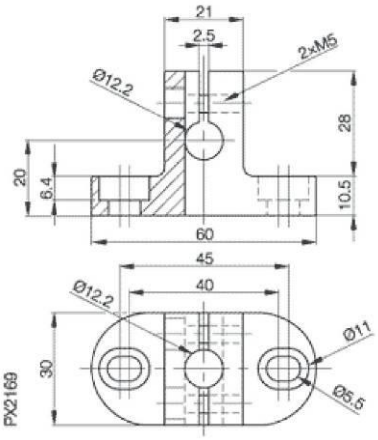


ST-5068
L-shaped fixing bracket for 90° mounting

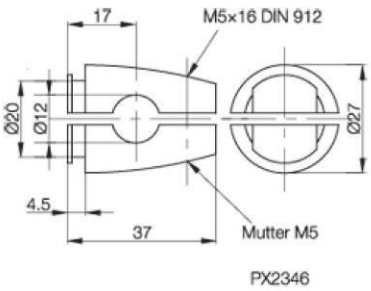


MOUNTING KIT

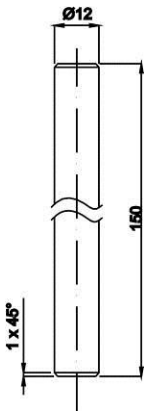
Base



Pin



Arm



MODEL	DESCRIPTION	ORDER N°
DataVS-ST-5068	L-shaped fixing bracket for 90° mounting	95A901320
DataVS-ST-5066	U-shaped fixing bracket for angle adjustment	95A901330
DataVS-CV-RJ45C-03	3m crossed Ethernet cable	95A901340
DataVS-CV-RJ45D-03	3m Ethernet straight cable	95A901350
DataVS-MK-01	Mounting Kit	95A901380
CV-A1-36-B-03	3m shielded cable with M12 8-pin connector	95A255430
CV-A1-36-B-05	5m shielded cable with M12 8-pin connector	95A255440
CV-A1-36-B-10	10m shielded cable with M12 8-pin connector	95A255450

GLOSSARY

Algorithm
A set of instructions that accomplish a task in a finite number of steps.

CMOS Image Acquisition Device
It consists of a set of photo-sensitive elements (→ pixels) arranged in a matrix on a chip surface that are exposed to light through a glass window.

Cross Cable
A special network cable where the transmitting wires and the receiving wires are crossed.

Focal Length
The distance between the rear nodal point of the lens, (the position at which light rays depart from the rear end of the lens), and the sensor.

FOV
Acronym for "Field Of View"; the field of vision of the sensor.

Gray Scale
The scale of values associated with the brightness of an image's pixels. For an 8-bit image, possible values are 256 (0 – 255, where '0' denotes black and '255' denotes white).

Hub
A node in a computer network that performs routing tasks.

Inspection
A process through which the sensor performs checks, measures certain features of the target image and returns a result.

Inspection Area
The shot area or part of the shot area that undergoes processing.

Inspection Result
can be:

"PASS", if the inspection (all controls/measurements) gave a positive outcome,
"FAIL", if the inspection (at least one control/measurement) gave a negative outcome or if the inspection could not be completed because one control/measurement was found to be impossible to perform.

IP Address
A unique address that identifies any one network device. It operates like a telephone number: to communicate with someone you need that person's phone number. In our case, you cannot communicate with the sensor unless you know its IP address. The IP address is made up of four numbers separated by dots. Example: 172.27.101.175.

LAN
Acronym for "Local Area Network" which identifies the network composed of computers connected together inside a defined area (for example an office or a building) that does not exceed the distance of a few kilometres. It includes interconnections and shared peripherals.

LED
Acronym for "Light Emitting Diode" which is a semiconductor device that produces photons from the recombination of electrons-electron holes.

Locator
A device that focuses processing on a definite portion of the inspection area.

Noise
Non-significant data that alters the source of information to be processed.

Operating Distance
The distance between the back of the lens and the target object.

Reference Image
A stored image that provides a reference for processing.

ROI
refers to a "Region Of Interest" (an area) within an image captured by the sensor, which is subject to processing/inspection.

Sample Image
see Reference Image.

Status
The result of the inspection in reference to an individual control/measurement (status may be "PASS" or "FAIL").

Target Image
An area of an acquired image that undergoes control/measurement.

Task
This refers to each individual action that, as part of a set of operations, leads to the solution of a problem.

DATALOGIC

www.datalogic.com