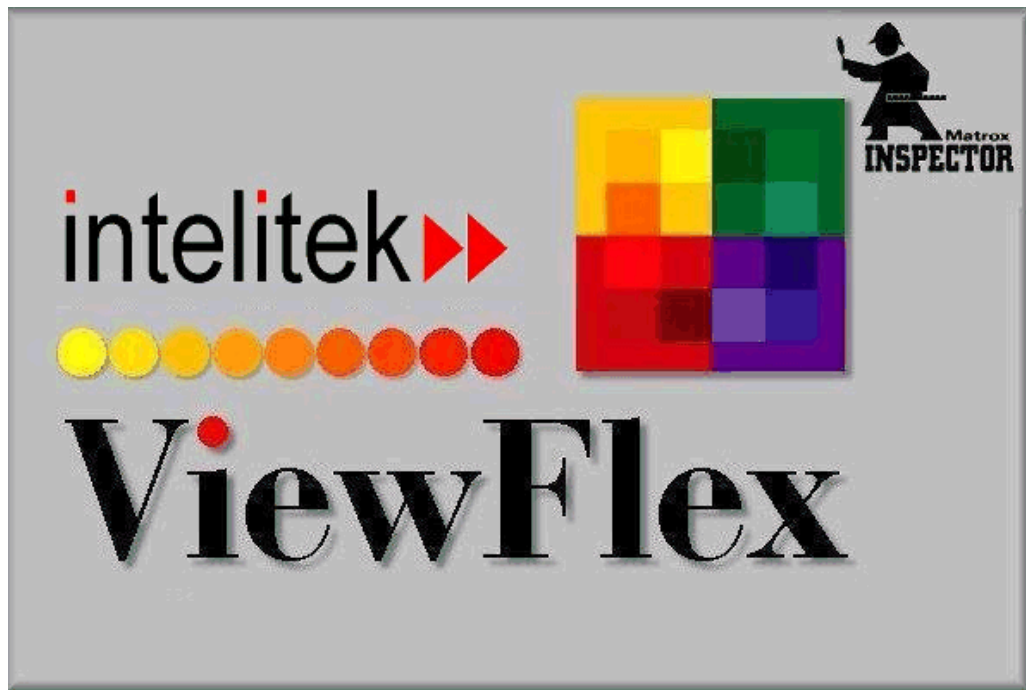


# ViewFlex

VERSION 3.3



USER MANUAL

*Catalog 100299 Rev. L*

*June 2016*

intelitek▶▶®

Copyright © Intelitek Inc.

ViewFlex User Manual

Cat.# 100299 Rev. L

June 2016

website: <http://www.intelitek.com>

email: [info@intelitek.com](mailto:info@intelitek.com)

Tel: (603) 625-8600

Fax: (603) 437-2137

Every effort has been made to make this book as complete and accurate as possible. However, no warranty of suitability, purpose, or fitness is made or implied. Intelitek Inc. is not liable or responsible to any person or entity for loss or damage in connection with or stemming from the use of the application and/or the information contained in this publication.

Intelitek Inc. bears no responsibility for errors that may appear in this publication and retains the right to make changes to the application and manual without prior notice.

The complete **Matrox Inspector User Guide** can be viewed and printed from the Matrox Inspector Help File `Inspector_UserGuide.chm` included on the ViewFlex software CD.

## Table of Contents

1.	ViewFlex Installation .....	1
1.1.	Unpacking the Equipment .....	1
1.2.	System Requirements .....	1
1.3.	Installing the ViewFlex System .....	1
1.4.	Connecting the Software Protection Key .....	1
1.5.	Installing the ViewFlex Software.....	2
1.6.	Connecting the Camera .....	9
3.	Viewing the ViewFlex Properties .....	10
1.7.	ViewFlex.ini File Configuration (for advanced users).....	11
1.8.	Accessing ViewFlex .....	12
1.8.1.	Running the ViewFlex Software .....	12
1.8.2.	Running ViewFlex for SCORBASE USB, USB Pro, ER5, ER7 and ER9 .....	12
2.	Software Operation .....	13
2.1.	ViewFlex for SCORBASE USB, USB Pro, ER5, ER7 and ER9 Toolbar .....	13
2.2.	Image Processing Tool .....	14
2.3.	Camera (Camera 1 / Camera 2 / ServeCam) .....	14
2.4.	Camera Options .....	16
2.4.1.	Configuration .....	16
2.4.2.	Image Format.....	18
2.5.	ServeCam Camera Connection (Client Camera only) .....	19
2.5.1.	Connect to ServeCam.....	19
2.5.2.	Results Table .....	19
2.5.3.	File .....	20
2.5.4.	Pattern Option .....	20
2.5.5.	View .....	20
2.6.	Vision Commands in SCORBASE .....	21
2.6.1.	SN Snap .....	21
2.6.2.	FO Find Object.....	22
2.6.3.	FB Find Blobs .....	22
2.6.4.	SP Set Position.....	23
2.6.5.	EF External Function.....	23
2.6.6.	GV Get Value .....	24
2.6.7.	CT Change Table.....	24
2.7.	Calibration .....	24

2.7.1.	Synchronizing the Vision System and the Robot .....	24
2.7.1.1.	Opening and Saving Calibration Files .....	26
2.7.1.2.	Example of Creating a Pattern Model .....	27
<b>3.</b>	<b>ViewFlex Toolbar .....</b>	<b>29</b>
3.1.	ACL Terminal for ViewFlex .....	29
3.1.1.	ACL ViewFlex String Commands .....	31
3.1.2.	ViewFlex Variable .....	31
3.2.	Abort All .....	33
3.3.	Go To Position .....	34
3.3.1.	Example of Go To Position Use .....	34
3.4.	Calibration .....	36
3.4.1.	Synchronizing the Vision System and the Robot .....	36
3.4.2.	Opening and Saving Calibration Files .....	38
3.5.	OpenCIM Device Driver .....	39
3.5.1.	ViewFlex.ini .....	40
3.5.2.	Operation Modes .....	40
3.5.2.1.	On-Line .....	40
3.5.2.2.	Manual .....	41
3.5.3.	Example of Script for Pass/Fail Test .....	42

# 1. ViewFlex Installation

## 1.1. UNPACKING THE EQUIPMENT

Before installing the equipment, check for signs of shipping damage. If any damage is evident, contact your freight carrier, and begin appropriate claims procedures. Make sure you have received all the items listed on the packing list. If anything is missing, contact your supplier.

## 1.2. SYSTEM REQUIREMENTS

For best performance, the following system is recommended:

- Dual Core with 3 GHz processor or higher, with CD drive
- At least 512 MB RAM (1 GB for Vista/Windows 7)
- A hard drive with at least 100 MB of free disk space (500MB free)
- Windows 32 and 64 bit:
  - Windows XP Professional
  - Windows Vista (Home Premium/Business/Ultimate editions)
  - Windows 7 (Home Premium and above)
  - Windows 8/8.1
  - Windows 10 (Pro/Education/Enterprise)
- A VGA or better graphics display, minimum 256 colors
- A mouse or other pointing device
- USB port

**Note:** Your operating system might have additional hardware requirements.

**Note:** Refer to the documentation for the Logitech C920-C or the Logitech website for the webcam's system requirements.

## 1.3. INSTALLING THE VIEWFLEX SYSTEM

Install the ViewFlex system in the following order:

1. Connect the software protection key. See *Connecting the Software Protection Key* on page 1.
2. Install the ViewFlex software. See *Installing the ViewFlex Software* on page 2.
3. Connecting the Camera. See *Connecting the Camera* on page 9.

## 1.4. CONNECTING THE SOFTWARE PROTECTION KEY

This step describes how to connect the software protection key, without this software protection key, the software is programmed to run on a 30-day trial basis.

To connect the software protection key, plug the software protection key into a USB port in your computer.



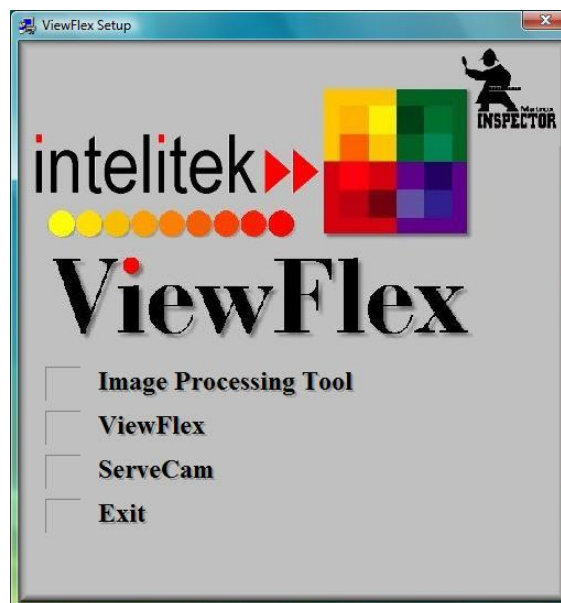
*Figure 1: Software Protection Key*

## 1.5. INSTALLING THE VIEWFLEX SOFTWARE

To Install the ViewFlex Software:

1. Insert the ViewFlex CD-ROM in the drive.
2. The installation procedure should begin to run automatically. If it does not, select **Start | Run**, and select **install\setup.exe**.

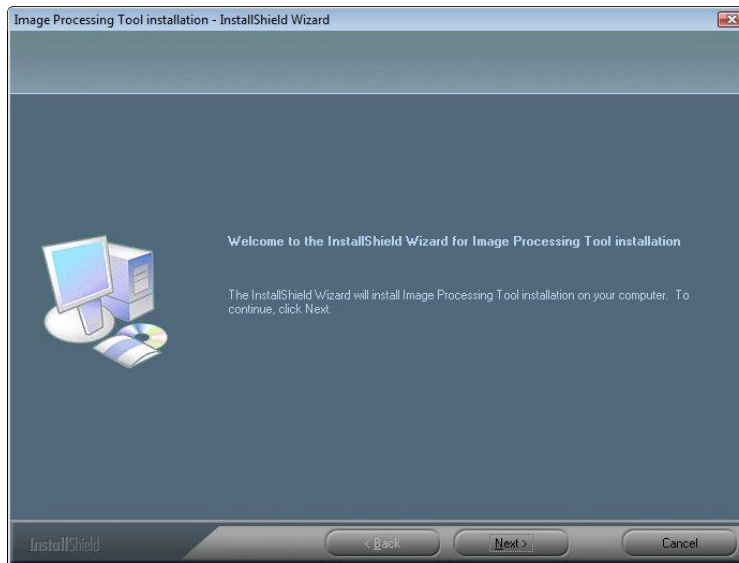
The ViewFlex Setup dialog box is displayed.



*Figure 2: ViewFlex Setup Dialog Box*

You must first install the **Image Processing Tool** module. Only then can you install the ViewFlex software.

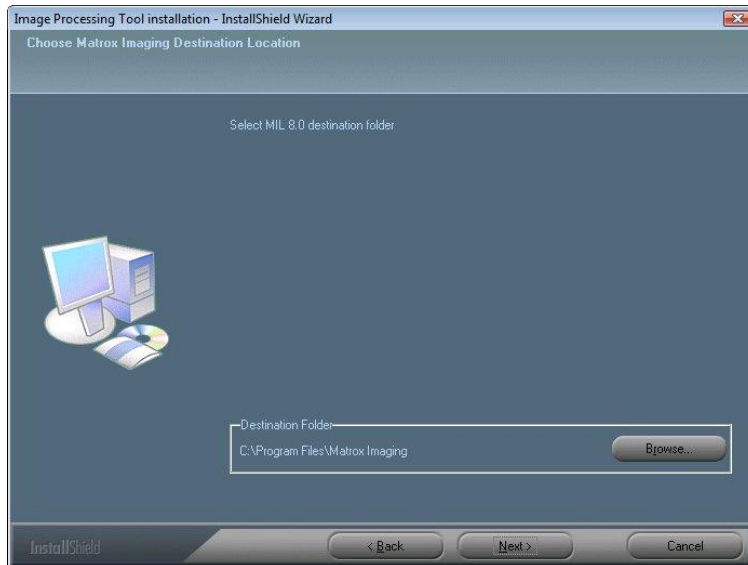
3. Click **Image Processing Tool**. The Image Processing Tool installation wizard is displayed.



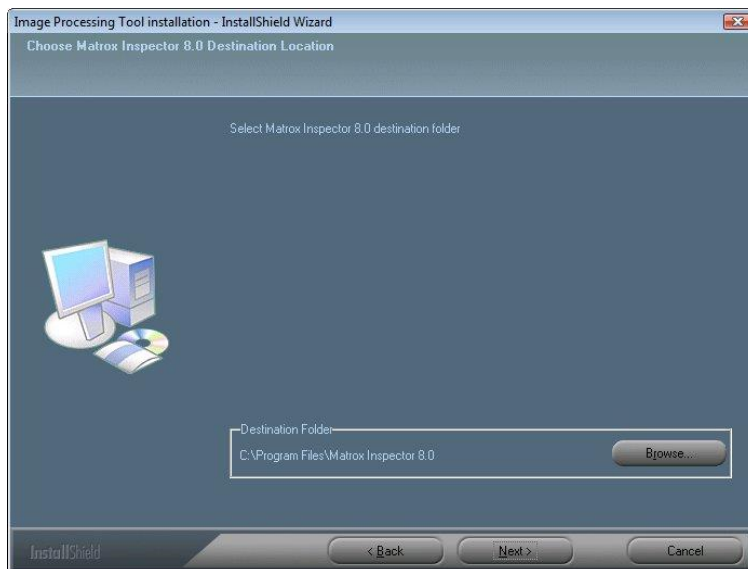
4. Click **Next**. The License Agreement window is displayed.



5. Select the top radio button to accept the terms of the license agreement and click **Next**. The Choose Matrox Imaging Destination Location window is displayed.

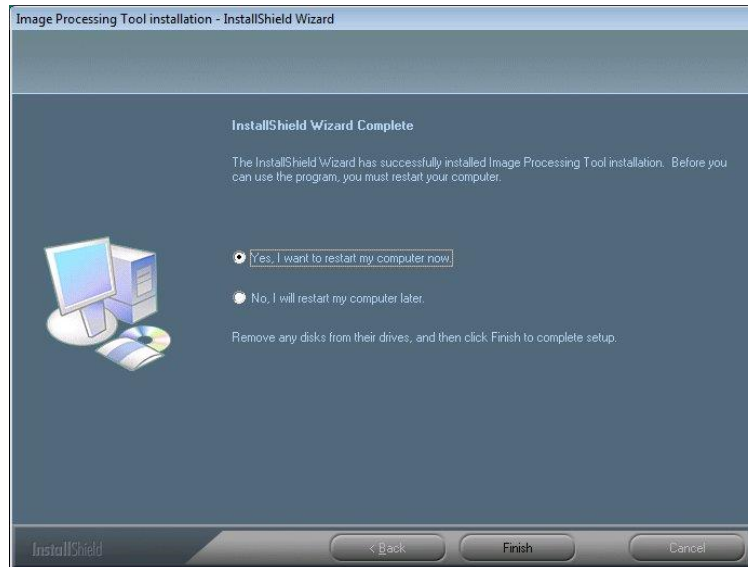


6. Click **Browse** and browse to the desired location for installing the Matrox Imaging Library, and then click **Next**. The Choose Matrox Inspector 8.0 Destination Location window is displayed.

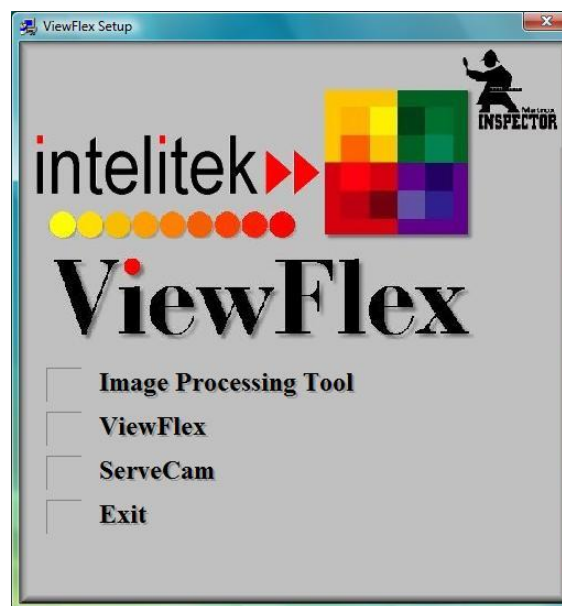




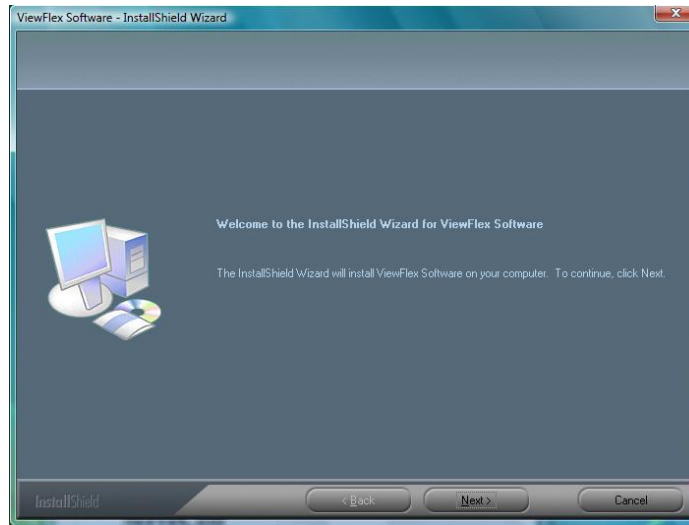
7. Click **Browse** and browse to the desired location for installing Matrox Inspector 8.0. Click **Next**. The software is installed and the InstallShield Wizard Complete window is displayed.



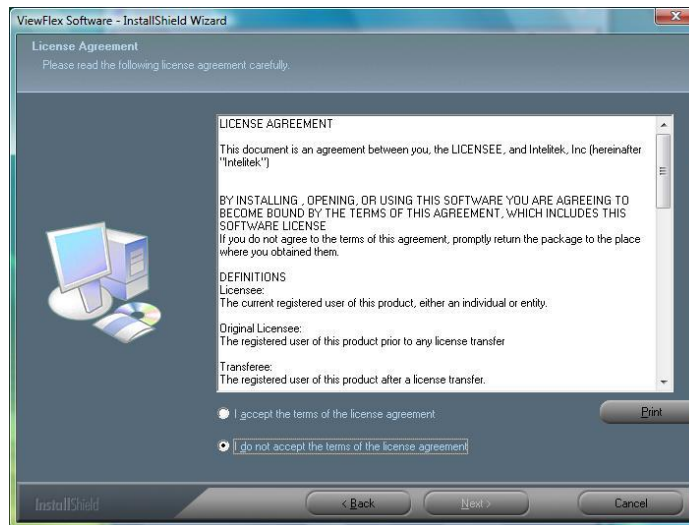
8. Select **Yes, I want to restart my computer now**.
9. Click **Finish**. The computer will restart.
10. Plug the software protection key into the USB port of your computer.
11. Start the ViewFlex installer by clicking on the icon of the CD drive in My Computer. The ViewFlex Setup window is displayed.



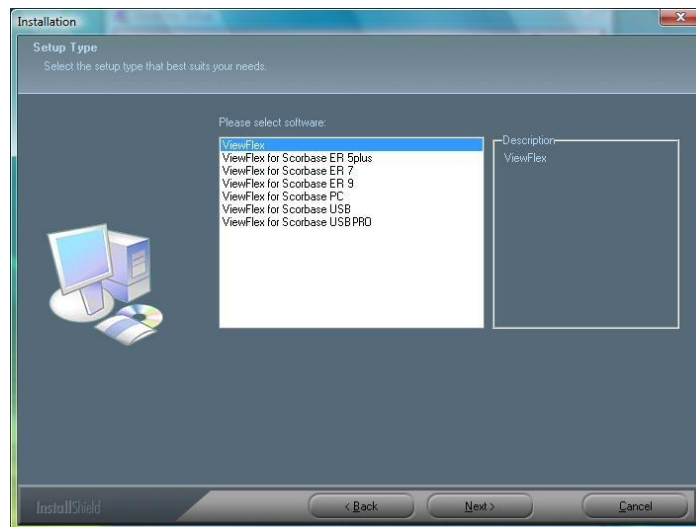
12. Select **ViewFlex**. The ViewFlex Software Wizard is displayed.



13. Click **Next**. The License Agreement window is displayed.



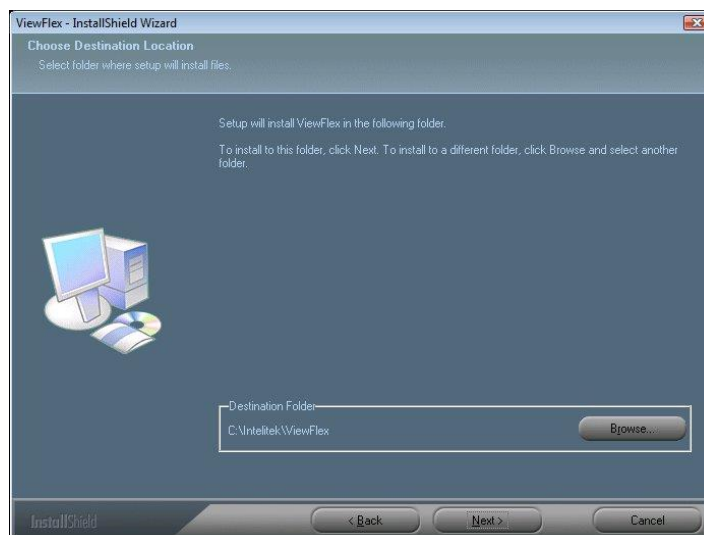
14. Select the top radio button to accept the terms of the license agreement and click **Next**. The Setup Type window is displayed.



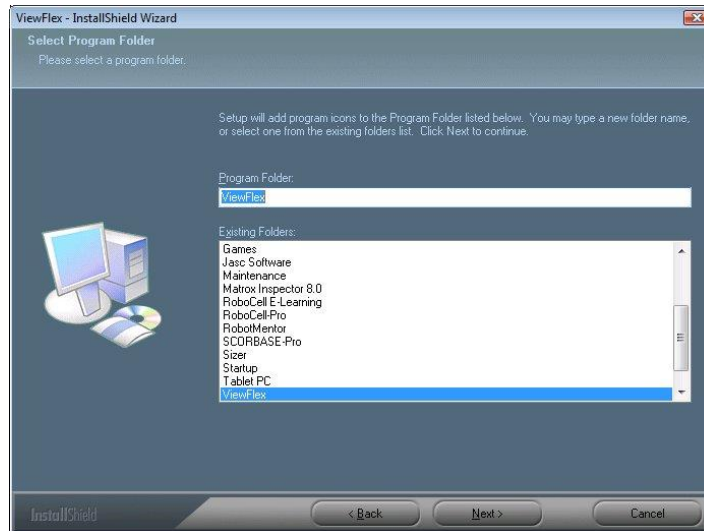
15. Select the required option, as follows:

- **ViewFlex:** Select this option to install ViewFlex for example, when using OpenCIM or when operating the robots via the ACL Controller.
- **ViewFlex for SCORBASE ER 5plus:** Select this option for ER-5.
- **ViewFlex for SCORBASE ER 7:** Select this option for ER-7.
- **ViewFlex for SCORBASE ER 9:** Select this option ER-9.
- **ViewFlex for SCORBASE USB:** Select this option for ER2U and ER4u.
- **ViewFlex for SCORBASE USB PRO:** Select this option for ER 9Pro.

16. Click **Next**. The Choose Destination Location window is displayed.



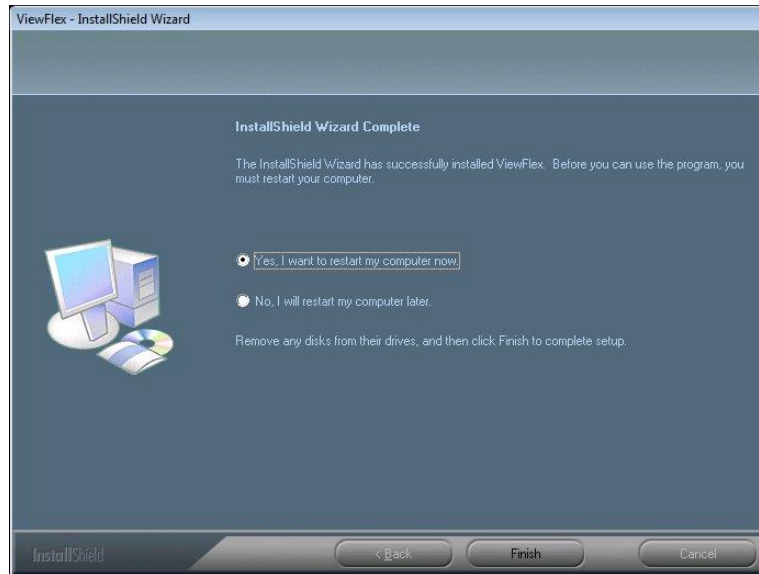
17. Click **Browse** and browse to the desired location for installing ViewFlex and click **Next**. The Select Program Folder window is displayed.



18. Select an existing folder or define a new one to add ViewFlex to and click **Next**. The software is installed. A prompt is displayed to create a desktop shortcut for ViewFlex.



19. Click **Yes** to create a Desktop shortcut click, or **No** to reject the prompt. The Installation Wizard Complete window is displayed.



20. Select **Yes, I want to restart my computer now.**

21. Click **Finish.** The computer will restart.

22. Plug the software protection key into the USB port of your computer.

**Note:** *If you receive the message: "Could not start ViewFlex" message, perform the following:*

1. From your windows Start menu select **All Programs.**
2. Right-click **Matrox Inspector.**
3. Select **Run as Administrator.** Wait until Matrox Inspector appears.
4. Exit Matrox Inspector.
5. Run ViewFlex again.

## 1.6. CONNECTING THE CAMERA

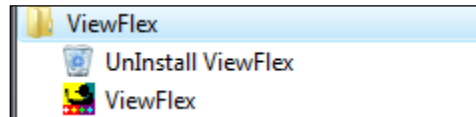
After installing the ViewFlex software, connect the Logitech C920-C webcam to an available USB port. The Windows drivers are loaded automatically.

## 3. Viewing the ViewFlex Properties

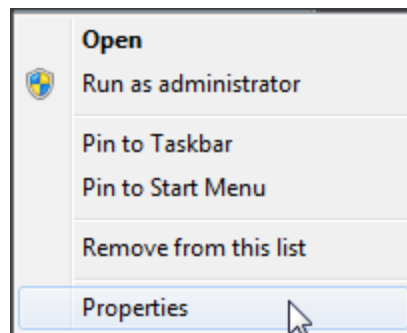
The ViewFlex Properties window enables you to view the ViewFlex Installation directory.

To view the ViewFlex properties:

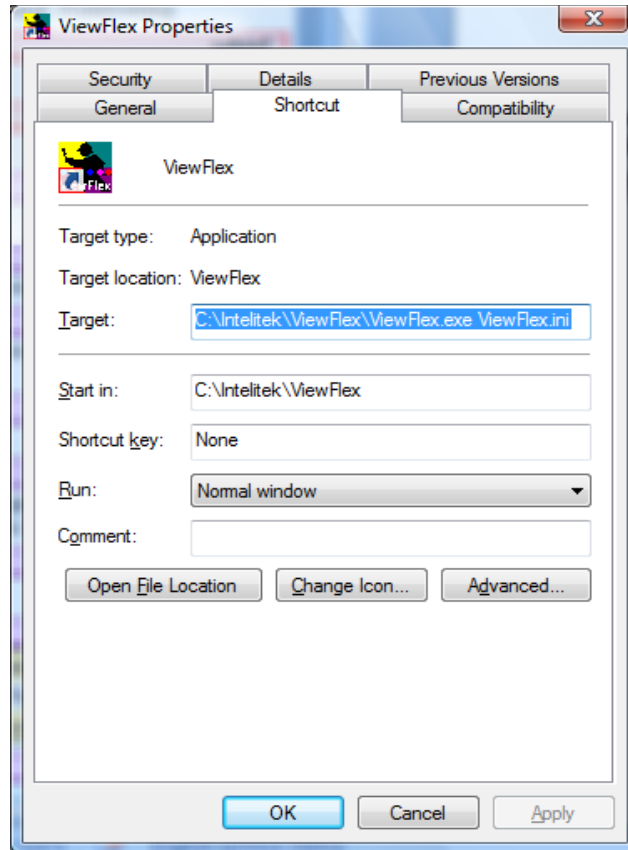
1. From the windows Start menu select **All Programs | ViewFlex**. The ViewFlex program folder is expanded.



2. Right-click  **ViewFlex**. A context menu is displayed.



3. Select Properties. The ViewFlex Properties window is displayed.



*Figure 3: ViewFlex Properties*

You can copy the Target path in the Shortcut tab and browse to that folder to access the other ViewFlex files.

## 1.7. VIEWFLEX.INI FILE CONFIGURATION (FOR ADVANCED USERS)

The ViewFlex.ini file initializes the ViewFlex software. The ViewFlex.ini file includes the following options:

- The directory in which the script files are located (only for the ViewFlex Device Driver):  
[Device Driver Definitions]

ScriptPath=\Vision script

- The controller type – A, B, PC, USB, USBPRO.  
[Controller Setting]

Controller=A

RobotType=ER14

④ (Note: You can choose: ER4, ER4u, ER5, ER9, ER9PRO, ER14, ER14PRO, viewSV)\

- Configuration of the second camera as either a local camera or a ServeCam:  
IsSecondCameraRemote = YES - camera 2 client servecam  
IsSecondCameraRemote = NO - camera 2 local camera

## 1.8. ACCESSING VIEWFLEX

### 1.8.1. Running the ViewFlex Software

To run ViewFlex, from the Start menu:

1. Select **Programs | ViewFlex**.
2. Click on the ViewFlex icon.

### 1.8.2. Running ViewFlex for SCORBASE USB, USB Pro, ER5, ER7 and ER9

The following explanation refers to SCORBASE for SCORBASE USB, USB Pro, ER5, ER7 and ER9:

To run ViewFlex for SCORBASE:

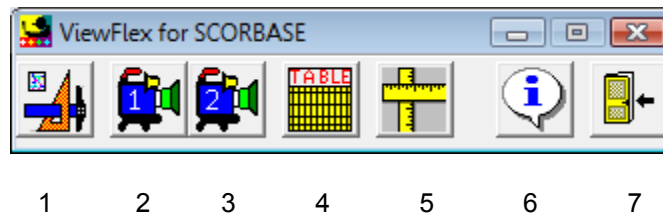
1. From the Start menu, select **Programs | ViewFlex for SCORBASE**.
2. Click on the ViewFlex for SCORBASE icon.



## 2. Software Operation

### 2.1. VIEWFLEX FOR SCORBASE USB, USB PRO, ER5, ER7 AND ER9 TOOLBAR

The ViewFlex for SCORBASE Toolbar contains icons that are shortcuts for selecting certain commands and functions:



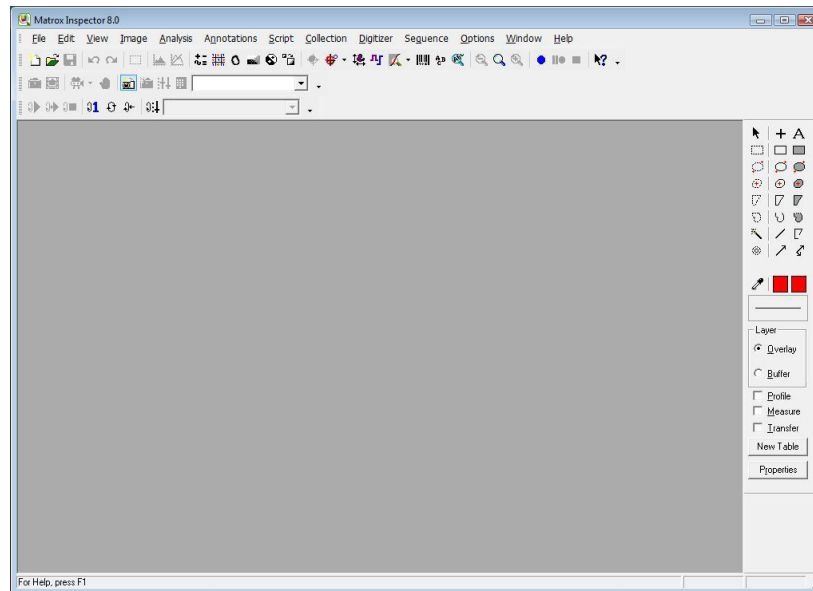
*Figure 4: ViewFlex for SCORBASE Toolbar*

1. **Image Processing Tool**; see details on page 14
2. **Camera 1**
3. **Camera 2**
4. **Results Table**
5. **Calibration**; see details on page 24
6. **About**
7. **Exit**

## 2.2. IMAGE PROCESSING TOOL



The Image Processing Tool icon opens the image-processing tool (*Matrox Inspector*).  
(Refer to the **Matrox Inspector Instruction Manual** for more information.)



*Figure 5: Matrox Inspector*

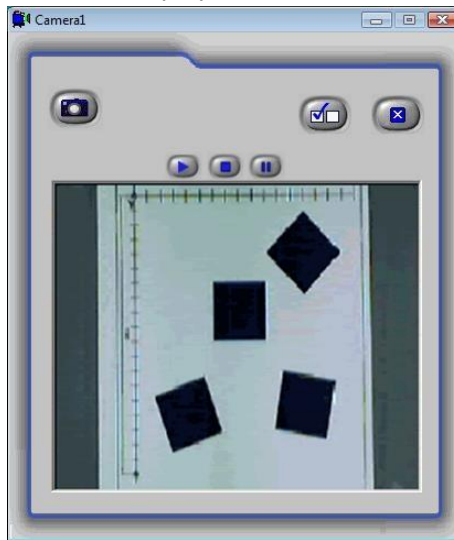
## 2.3. CAMERA (CAMERA 1 / CAMERA 2 / SERVECAM)



When two cameras are used, you will be prompted to select a camera to use.  
The camera icons open the Camera (capture) window or the Client Camera window.  
The Client (remote) camera operation is as follows:

- When the Client Camera is activated, it connects to the server.

The Client Camera displays the IP address of the computer that the ServeCam is connected to.



*Figure 6: Camera Window*



*Figure 7: Camera Window*



**Play**

Enables camera streaming.



**Stop**

Stops camera streaming.



**Pause**

Pauses camera streaming.



**Snapshot (ALT + S)**

Grabs (captures) a picture and displays it in a frame.

OR

Inserts an updated picture from the ServeCam.



**Options (ALT + O)**

Opens a dialog box use for video settings and configuration.

**Connection (available only in Client Camera)**

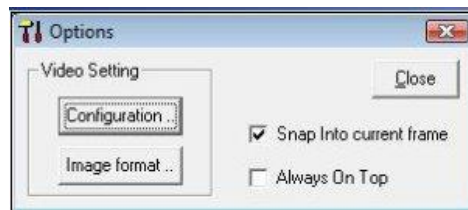
Enables connection of remote cameras on a network.



**Exit**

Closes the Camera window.

## 2.4. CAMERA OPTIONS



*Figure 8: Camera Options*

### Always On Top

Keeps the Camera window above all other windows.

### Snap Into Current Frame

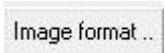
The captured image will be put into the current frame in the Image Processing Tool. The part of the image that will be inserted into the frame depends on the current frame size. If left unchecked, a new frame will be created each time an image is captured (every Snap).

### Configuration




Displays the camera configuration dialog box. There is a different configuration dialog box for each camera, as described in the Configuration section below.

### Image Format



Displays the Video Format Properties dialog box, described in the following section.

### 2.4.1. Configuration

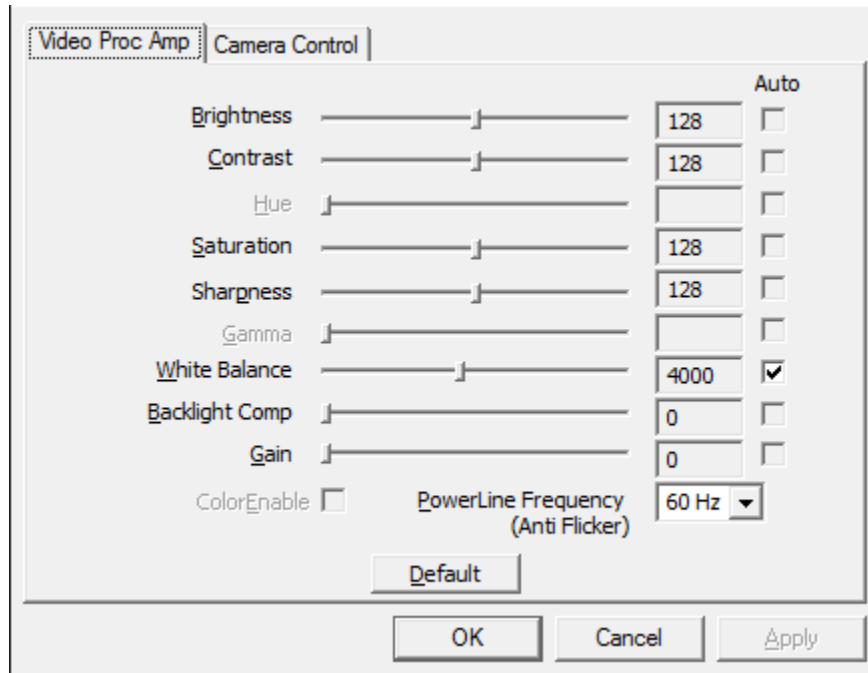
The Video Source Properties dialog box is displayed by clicking the  button in the Camera Options dialog box.

The Video Source Properties window contains the following configuration tabs:

- **Video Proc Amp Tab**, contains device settings.
- **Camera Control Tab**, contains camera control options, such as zoom, focus, exposure and more.

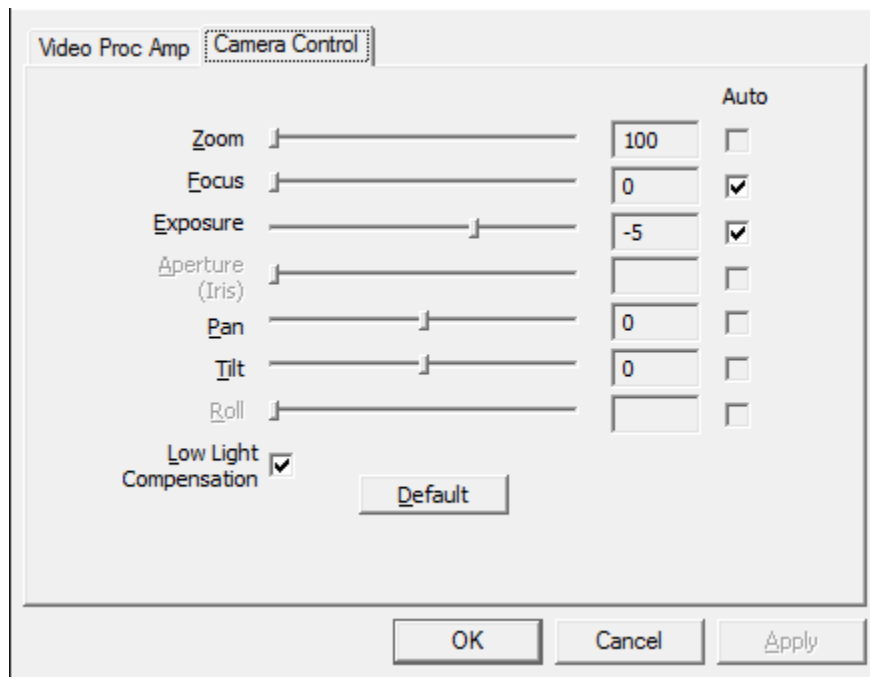
#### 2.4.1.1.1. Tabs description

- Video Proc Amp Tab




The Video Proc Amp tab enables you to define your device settings, by dragging the relevant sliders such as Brightness, Contrast, Hue, Saturation and more. You can define your default settings as required and then click **Default** to save them.

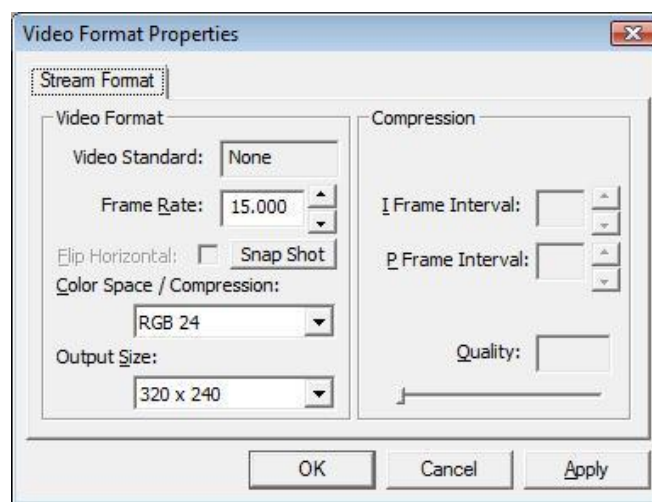
- Camera Control Tab



The Video Proc Amp tab enables you to define your camera control settings, by dragging the relevant sliders such as Zoom, Focus, Exposure and more. You can define your default settings as required and then click **Default** to save them.

### 2.4.2. Image Format

The Video Format Properties dialog box is displayed by clicking the  button in the Camera Options dialog box.



*Figure 9: Video Format Properties Dialog Box*

The Format Properties dialog box enables you to define video format settings, such as frame rate, color space, output size and more.

## 2.5. SERVECAM CAMERA CONNECTION (CLIENT CAMERA ONLY)

### 2.5.1. Connect to ServeCam

If there is more than one ServeCam on the network, then you can make a connection to the specific ServeCam that you want by entering the IP address of that ServeCam's computer.

The default IP address is the one that initially appears in the box, which was automatically detected.

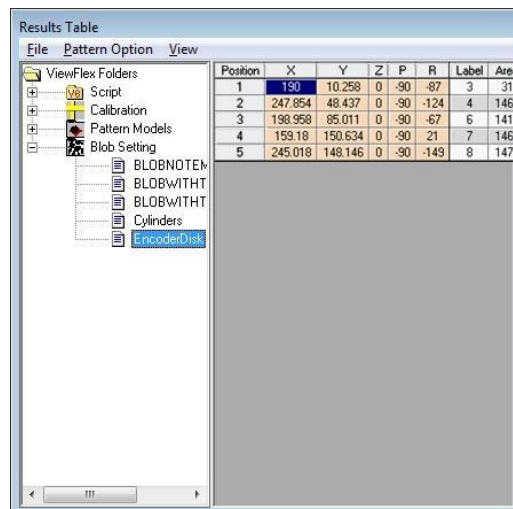
Clicking OK will connect to the ServeCam at the IP address shown.

To configure a second local camera in place of a ServeCam Camera, refer to *ViewFlex.ini File Configuration (for advanced users)* on page 11.

### 2.5.2. Results Table



The Results Table window is divided into two sections: one section is a tree representing the ViewFlex folders, and the second section is a table with the coordinates of the object in the robot world – including blob features.



Position	X	Y	Z	P	R	Label	Area
1	150	10.258	0	-90	-87	3	31
2	247.854	48.437	0	-90	-124	4	146
3	198.958	85.011	0	-90	-67	6	141
4	159.18	150.634	0	-90	21	7	146
5	245.018	148.146	0	-90	-149	8	147

Figure 10: Results Table

Every time a new file is saved in one of the ViewFlex for SCORBASE folders, the tree is updated.

- Selecting one of the **Pattern Models** executes Find Object with the model on the current image.
- Selecting one of the **Blob Settings** executes Find Blobs with the Blobset on the current image
- Selecting one of the **Scripts** executes the first function/ subroutine in the Script.

### 2.5.3. File

**Save Results** to a file so that they can be opened with Excel.

**Load results** from a file into the Results Table.

### 2.5.4. Pattern Option

**Foreground:** Defines whether black pixels or white pixels are to act as the foreground (or blob) pixels.

### 2.5.5. View

**Always on top:** Checking this box prevents this window from being hidden by other windows.

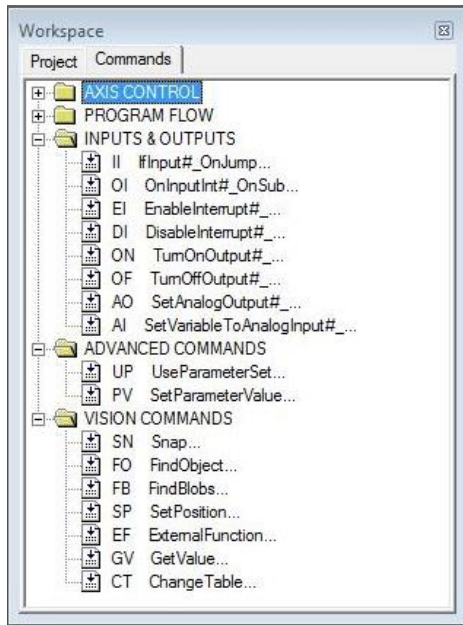


## 2.6. VISION COMMANDS IN SCORBASE

All SCORBASE programming commands are available from the Command List and are easily compiled into a robot program.

The list shows the two-letter hot-key combinations that allow you to enter commands from the keyboard.

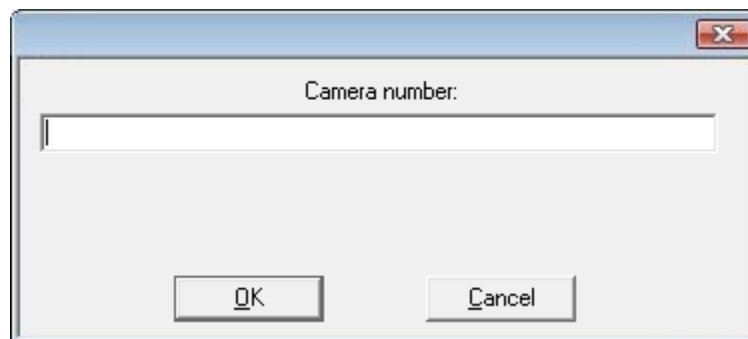
Many commands open dialog boxes for completing the command line parameters.



*Figure 11: Vision Command Group*

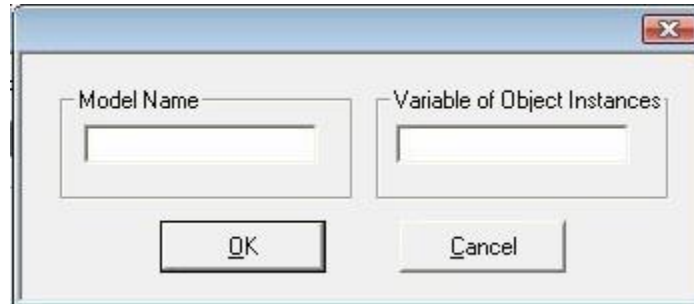
### 2.6.1. SN Snap

Captures an image.



### 2.6.2. FO Find Object

Finds an object according to the Pattern Model Name that has been saved in the Pattern folder. Returns the number of instances of the object.



Enter the name of the model in the **Model Name** field.

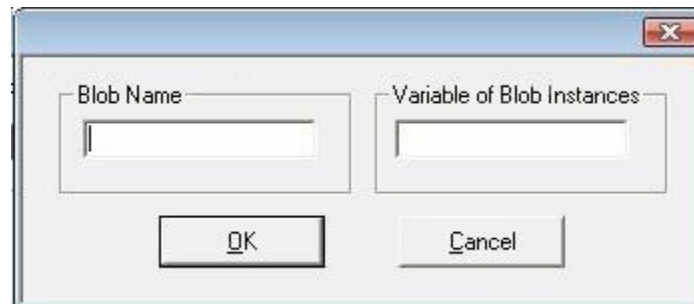
① (Note: The name must be the same as it is in the Result Table.)

The SCORBASE variable must be entered into the **Variable of Object Instances**. The results (Number of Object) will be assigned to the variable.

All of the robot's coordinates and blob features will be put into the Results Table.

### 2.6.3. FB Find Blobs

Finds a blobs according to the blob analysis name that has been save in the Blobs folder. Returns the number of instances of the Blob.



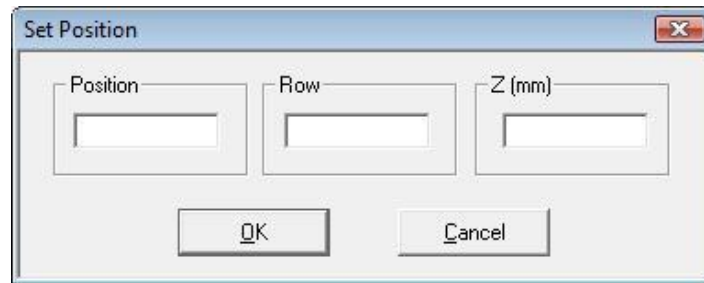
Enter the name of the model in the **Blob Name** field. (*The name must be the same as it is in the Results Table.*)

The SCORBASE variable must be entered into the **Variable of Blob Instances** field. The results (Number of Blobs) will be assigned to the variable.

All of the robot's coordinates and blob features that were defined by the user will be put into the Results Table.

#### 2.6.4. SP Set Position

Sets the coordinate value from the Results Table into the SCORBASE Position Number.



In the **Position** field, insert the SCORBASE Position Number.

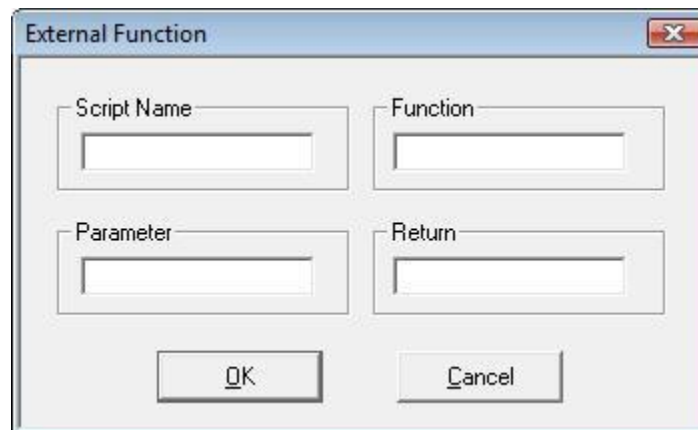
In the **Row** field, insert the Position Number (Row) in the Results Table.

In the **Z (mm)** field insert the Z-coordinate. By default the value in the Z-coordinate will be the same as the value of the calibrated position.

Z can be a variable or constant.

#### 2.6.5. EF External Function

Executes the function or subroutine that the user writes in the Matrox script.



The results of the function will be assigned to the variable.

In the **Script Name** field, insert the name of the script in the script folder.

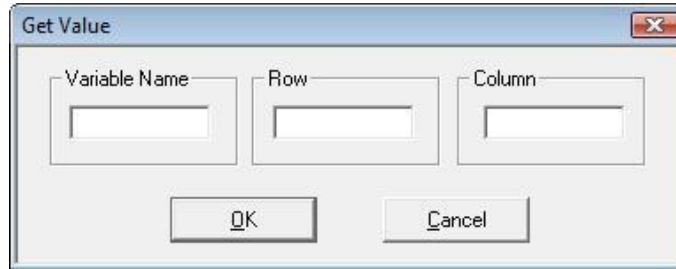
In the **Function** field insert the function or subroutine name that is in the script.

In the **Parameter** field insert the parameter that can be the SCORBASE variable or constant. (*Note: Available for ViewFlex for SCORBASE for ER-4u only.*)

In the **Return** field, insert SCORBASE variable that will get the function return.

### 2.6.6. GV Get Value

Retrieves a value from any cell in the Results Table.



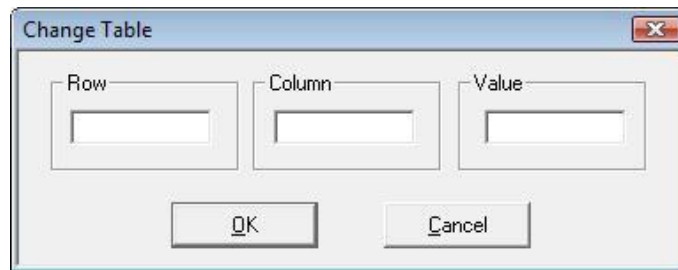
In the **Variable Name** field, insert the SCORBASE Variable. The cell value will be assigned in the variable.

In the **Row** field, insert the cell row in the Results Table.

In the **Column** field, insert the column name in the Results Table. The **Column** can be a number or the exact column name.

### 2.6.7. CT Change Table

Changes the value of any cell in the Results Table.



In the **Row** field, insert the cell row in the Results Table.

In the **Column** field, insert the column name in the Results Table. The **Column** can be a number or the exact column name.

In the **Value** field, insert the SCORBASE variable or constant that you want the cell to have.

## 2.7. CALIBRATION

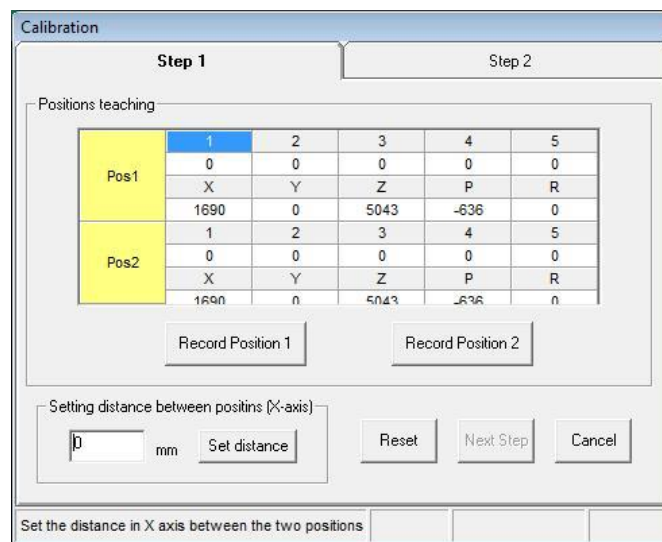


### 2.7.1. Synchronizing the Vision System and the Robot

To calibrate real-world measurements with the vision system's system of pixels, perform the following steps:

1. Execute the HOME routine.

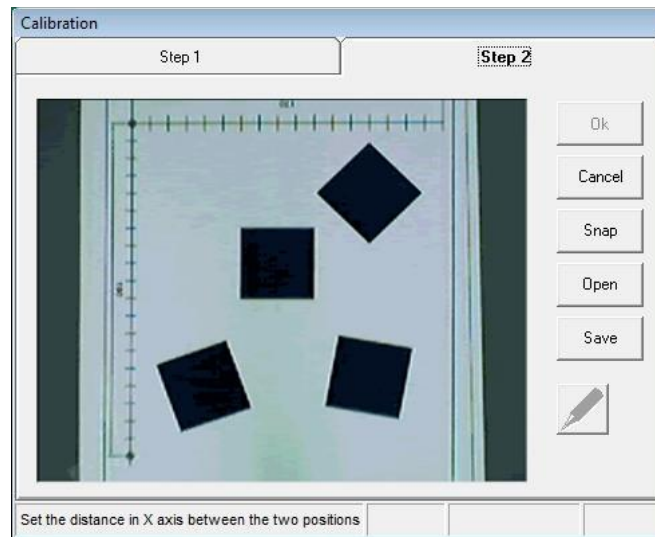
2. Open the Client Camera.
3. Check the camera to make sure that the picture is adjusted, clear and free of distortions.
4. Click the Calibration icon.
5. Using SCORBASE, move robot arm into desired position within the camera's field of vision.
6. Using a pen or marker, mark a position on a piece of paper (or on the coordinate grid) close to the robot and strong enough to be seen in the camera picture. This position should be aligned with the TCP (tool center position – the exact center of the robot gripper). This first position is called Pos 1.
7. Click **Record Position 1**.



**Figure 12:** Calibration Step 1 Dialog Box

8. Along the X-axis only, move the robot to a position on the X-axis that would be greater (+X) than Pos 1. Take a writing instrument and mark the position. This second position is called Pos 2.
9. Click **Record Position 2**.
10. Now you will see in the Combo box the distance between the two positions (Setting Distance) on the X-axis. To achieve the most accurate measurement, take a ruler and measure the distance between Pos 1 and Pos 2 by hand.
11. Change the distance if necessary according to the hand measurement.
12. Click **Set Distance**.

13. Click **Next Step**. The Step 2 tab is displayed.



*Figure 13: Calibration Step 2 Dialog Box*

14. If necessary, move the robot arm out of the camera's view.

15. Click **Snap** in the Step 2 dialog box.

16. You will be able to recognize Pos 1 and Pos 2 in the image.

- Click the Pen icon. The Pen icon will start flashing after being engaged. Your mouse arrow will become a cross.
- Click on Pos 1, and then make a dragging motion, using your mouse, toward Pos 2. This movement represents an increase in the X-axis. (Shows the direction of increase of the X-axis).

17. Click the Pen icon again. The X and Y-axes are now set and the distances are in real-world millimeters.

When you move your mouse along the image, Pos 1 becomes coordinate 0,0 (center-point) and the directions of X and Y are now the same as the robot. The movement of the mouse along the axes shows changes in X and Y.

18. Click **Save** to save the calibration.

19. Browse to C:\Intelitek\Viewflex for Scorbases USB\Calibration. Define a file name and click **Save**.

20. Click **OK**. The Vision System is calibrated and synchronized with the robot, and all new images are calibrated.

### 2.7.1.1. Opening and Saving Calibration Files

The Configuration Step 2 dialog box enables file management operations.

## Save

Saves the calibration. The filename extension is CAL.

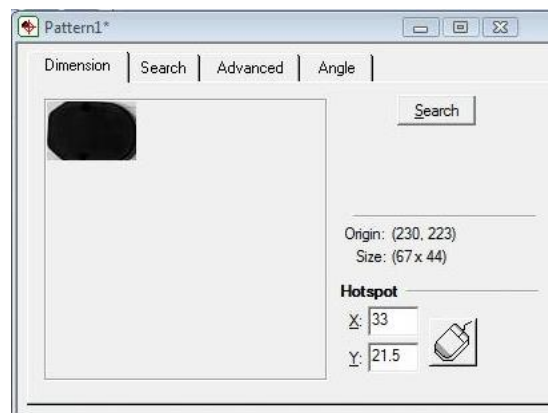
## Open

Allows you to load backup calibration files. When the OK icon is clicked, the system is calibrated.

You can configure immediately in the Step 2 tab and open the calibrated file.

### 2.7.1.2. Example of Creating a Pattern Model

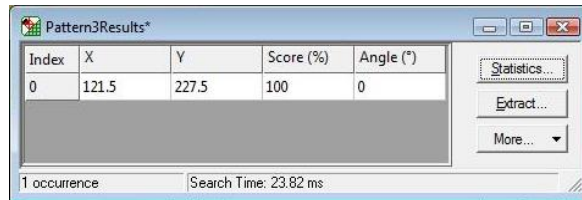
1. The system must be calibrated so that real-world measurements correspond to the vision system's set of pixels.
2. Open the Client Camera.
3. Click the Image Processing Tools icon to open Matrox Inspector.
4. For this example, set a cube down within the camera's field of vision, so that its edge is parallel to the edge of the Client Camera window.  
In order for both the robot gripper and the cube to be set to zero degrees, the cube must be defined as parallel to the X-axis of the camera's world. Subsequently, the robot arm's rotation will be relative to this point.
5. In Client Camera, click **Snap**.
6. Click the ROI (Region of Interest) icon, and outline the perimeter of the cube that was set.
7. Click the Pattern Matching icon.



*Figure 14: Pattern Matching Model Dialog Box*

- From the Dimension tab, you will see the cube that was outlined using the ROI icon.
- From the Search tab, click **All**.
- From the Angle tab, check Enable Search With Rotate. Then enter 180 for Delta Negative, and 180 for Delta Positive (this enables a full 360-degree search).

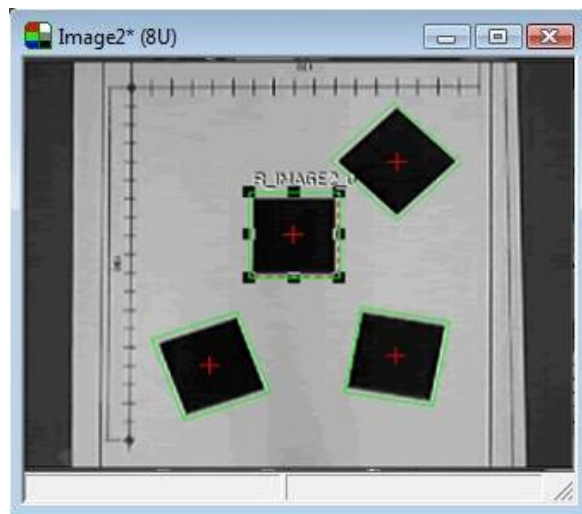
- Click **Search**. This will build a Measurement Table containing all the positions of the object relative to Pos 1.



Index	X	Y	Score (%)	Angle (°)
0	121.5	227.5	100	0

1 occurrence      Search Time: 23.82 ms

*Figure 15: Measurement Table*



*Figure 16: Searched Images*

8. Save the Pattern Matching Model in the Pattern Folder, located in the ViewFlex for SCORBASE root folder. Ensure that the Pattern dialog box is selected when saving.
9. Open the Results Table and expand the Pattern Models in the folder tree. Check that the name you gave is listed in the tree.
10. Use this name when you Find Object.



## 3. ViewFlex Toolbar

The ViewFlex Toolbar contains additional icons to the ViewFlex for SCORBASE USB Toolbar, and some of the icons that are the same activate certain commands and functions with different functionality than that of ViewFlex for SCORBASE USB:



*Figure 17: ViewFlex Toolbar*

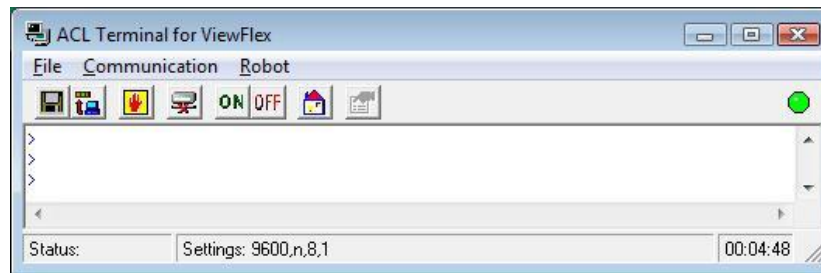
1. Image Processing Tool
2. Camera 1
3. Camera 2
4. ACL Terminal (for ViewFlex)
5. Abort All
6. Go Position
7. Calibration
8. OpenCIM Device Driver
9. About
10. Exit

### 3.1. ACL TERMINAL FOR VIEWFLEX



ACL, Advanced Control Language, is an advanced, multi-tasking robotic programming language.

The ACL Terminal for ViewFlex is the software interface that provides access to the controller from a PC, and provides functions needed to configure, program and operate the robot system.



*Figure 18: ACL Terminal for ViewFlex*



**Save**

Saves a listing of all user programs, variables and positions in one file. The files are in ACL format (.acl).



**Download**

Downloads data from a user-backup file in the host computer to the controller. (Does not erase or modify existing programs.)



**Abort All**

Immediately aborts all running programs and stops movement of axes.



**On-Line/Off-Line**

Opens/closes the communications port.



**ON-CON**

Enables servo control for all axes, a specified group of axes, or a single axis.



**OFF-COFF**

Disables servo control for all axes, a specified group of axes, or a single axis.



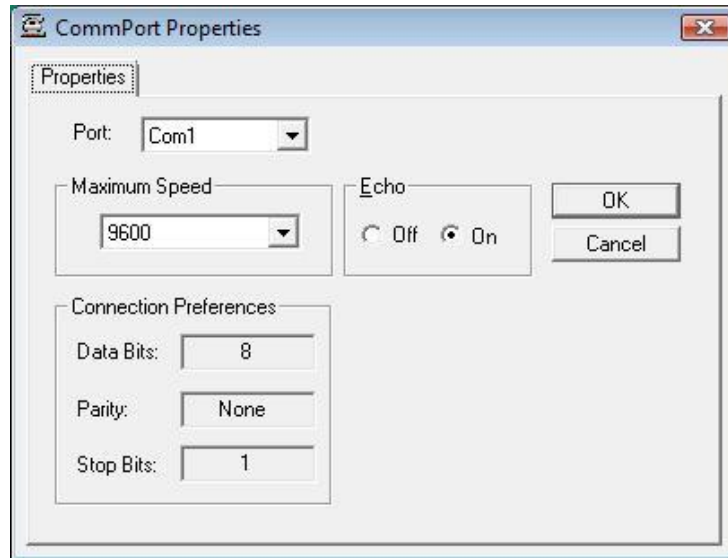
**Home**

Drives all robot axes to their home position.



## Port Properties

Opens a dialog box that allows you to configure the RS232 serial communication port in the PC that communicates with the robot controller:



*Figure 19: ACL Terminal CommPort Properties*

### 3.1.1. ACL ViewFlex String Commands

The ViewFlex String Commands are written as:

**Print / Println “%ViewFlex string commands %”**

ViewFlex helps you make applications that combine vision and robot guiding that will use the ACL language for controlling and executing robot tasks.

### 3.1.2. ViewFlex Variable

Vision functions are written in Visual Basic (VB) script. The vision script will operate the vision task. To synchronize the vision and robot tasks, there is the Global Variable (VF). After executing the Vision Function or subroutine, the VF value will be 1 (VF=1).

#### SNAP

**Format:** %SNAP%

**Description:** Will capture an image from the camera and put it into the Image Processing Tool.

**Examples:** PRINTLN "%SNAP%"

#### RUN

**Format:** %RUN prog%

**Description:** Where: prog is the name of a function or subroutine in the script. The function returns the ACL command back to the controller.

**Examples:** PRINTLN "%RUN PROG%"

This executes the function or subroutine in the open VB script file. If you execute the function, the results will be sent to the controller.

*The script must be loaded before it can be executed from the ACL.*

## SETPOS

**Format:** %SETPOS Position, Table index, Table name, Z(mm)%

**Description:** Where:

Position is a defined robot position.

Table index is a position of the object in a measurement table that was found in the image after searching. The reserve word ALL means that if a Position is the name of a vector all the positions in the table will be set.

Table name is the name of the measurement table. If omitted, it means Current table.

Z is the value of the Cartesian coordinate Z in millimeters.

**Examples:** PRINTLN "%SETPOS pos,1, , 60%"

Position pos receives the coordinate value of position 1 in the Current measurement Table, while the coordinate value of Z is 60 mm.

PRINTLN "%SETPOS pv[4],3,table1 , 80%"

Position pv[4] receives the coordinate value of position 3 in the measurement table named table1, while the coordinate value of Z is 80 mm.

PRINTLN "%SETPOS pv,ALL,table2 , 80%"

Vector pv receives the coordinate values of ALL positions in the measurement table named table2, while the coordinate value of Z is 80 mm.

**Project:** The following example project shows how the ACL program is used with the Vision System, in order to automate the robot and the Vision System. The example is made from two programs:

ACL: for the robot task

Visual Basic Script: for the vision task.

GOPOS.ACL

(Note: N should be a Global Variable.)

The code initiates the global variable VF before sending comments to the Vision System, so that it can cause the ACL program to wait until the vision command is finished. This happens when VF is 1.

The program tells the Vision System to Snap, then executes the GetPos function in the script. The results of the GetPos function is a string that is an ACL command. If the search results are 3, for example, then set N to 3 ("set n=3").

SETPOS takes the position of the object from the measurement table that was found after the search, and converts it to positions that can be sent.

ALL takes every position from the Measurement Table and sets them into positions in the vector.

At the end of GoPos, the program moves the robot to the position in the vector.

```
PROGRAM GOPOS
*****
SET VF = 0
PRINTLN "%SNAP%"
WAIT VF = 1
SET VF = 0
PRINTLN "%RUN GETPOS%"
WAIT VF = 1
SET VF = 0
PRINTLN "%SETPOS POS,ALL,,60%"
WAIT VF = 1
FOR I = 1 TO N
  MOVED POS[I]
ENDFOR
END
```

### GETPOS.BAS

In order for the Run command to work, the script must be loaded.

The model for the object whose script will be searched should also be loaded.

When the script will be executed, a measurement table with the positions will be created. The script will also send a command that initiates a sign of the number of objects that were found.

```
Function GETPOS()As String
  I_IMAGE1$ = Insptr.ImgGetCur
  Insptr.ImgSetCurrent I_IMAGE1$, R_Def$, ALL_BANDS
  Insptr.ImgConvertType(TO_8U)
  Insptr.MeasNew()
  M_X_MOD$=Insptr.PatGetCur
  Insptr.PatSetCur M_X_MOD$
  x=Insptr.PatFind
  Insptr.ImgClose
  Num=Insptr.MeasGetLastID
  GETPOS="SET N=" + CStr(num)
End Function
```

## 3.2. ABORT ALL



**Format:** [Ctrl] + A

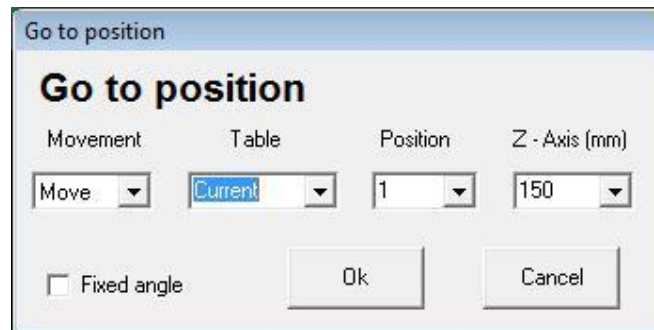
**Description:** Immediately aborts all running programs and stops movement of axes.

[Ctrl] + A is the fastest software method for stopping program execution and halting

movement of all axes. It can be used at any moment, even while entering another command, in order to instantly halt programs and axes.

*This command is entered from keyboard, but does not require [Enter] for execution.*

### 3.3. GO TO POSITION



*Figure 20: Go to Position Dialog Box*

#### **Movement**

Move/MoveL (Move Linear).

#### **Table**

Can be the Current table, or any other named measurement table.

#### **Position**

The index of a position in the measurement table.

#### **Z-axis (mm)**

The Cartesian coordinates in millimeters.

#### **Fixed Angle**

Enabling this option causes only the X and Y axes to be. This option can be used with circular objects when the gripper rotation is not important.

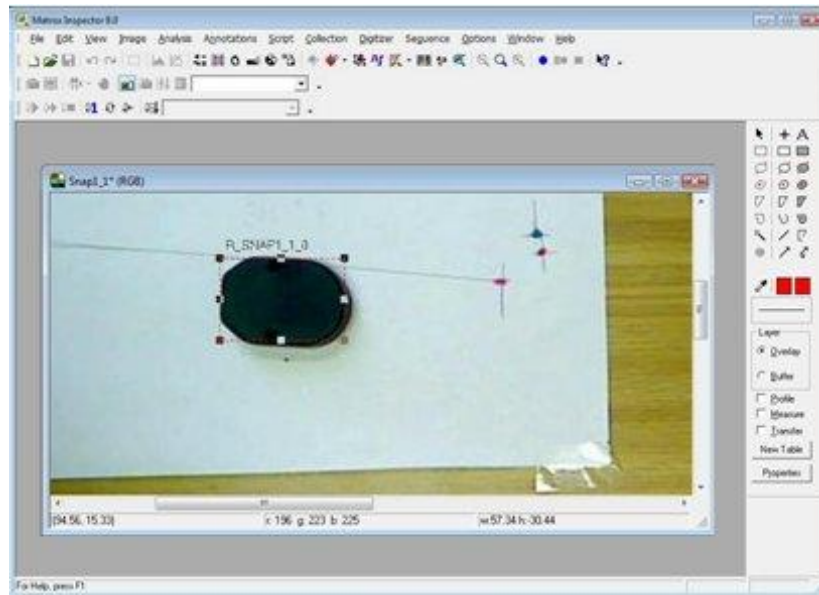
#### **3.3.1. Example of Go To Position Use**

System must be calibrated, so that real-world measurements correspond with the Vision System's world of pixels.

1. Open Camera 1.
2. Click the Image Processing Tool icon.
3. In this example, put a cube down within the camera's field of vision, so that its edge is parallel to the edge of Camera 1 dialog box's border.  
In order for both the robot gripper and the cube to be set to zero degrees, the cube must be

defined as parallel to the X-axis of the camera's world. Subsequently, the robot arm's rotation will now be relative to this point.

4. In Camera 1, click **Snap**.
5. Click the ROI (Region of Interest) icon, and outline the perimeter of the cube that was set.
6. Click the Pattern Matching icon.



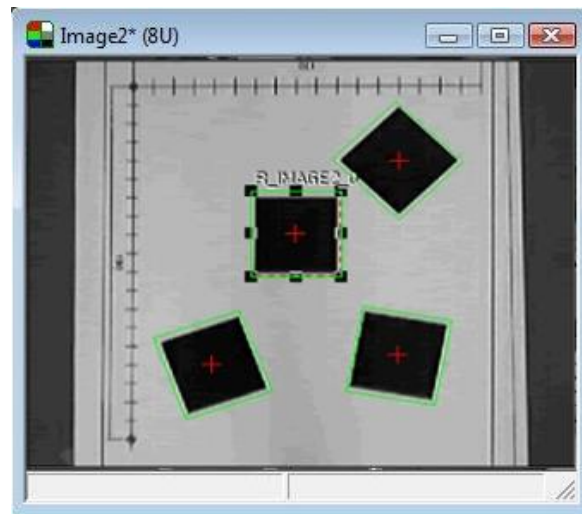
*Figure 21: Pattern Matching Model Dialog Box*

7. From the Dimension tab, you will see the cube that was outlined using the ROI icon.
8. From the Search tab, click **All**.
9. From the Angle tab, check Enable Search With Rotate. Then enter 180 for Delta Negative, and 180 for Delta Positive (this enables a full 360-degree search).
10. Click **Search**. This will build a Measurement Table containing all the positions of the object relative to Pos 1.

Index	X	Y	Score (%)	Angle (*)
0	121.5	227.5	100	0

1 occurrence      Search Time: 23.82 ms

*Figure 22: Measurement Table*



*Figure 23: Searched Images*

11. In this example, choose the Current table (which will be the opened table), and Position 1 (Pos 1).
12. Click **OK**. The robot will move according to the Go To Position and find the cube.

## 3.4. CALIBRATION



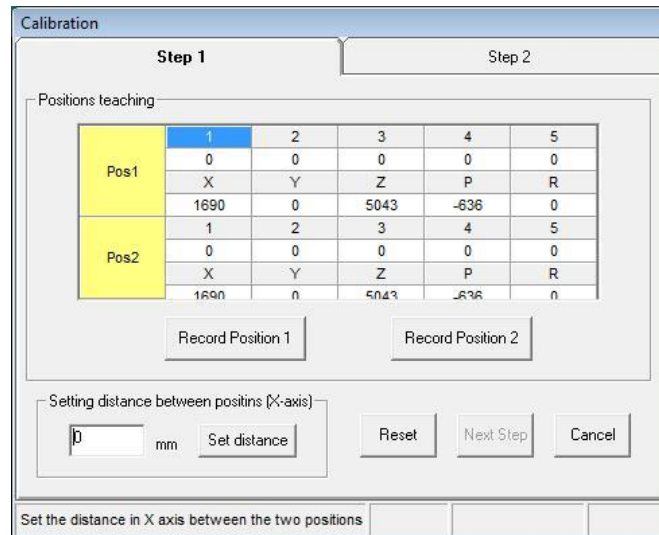
### 3.4.1. Synchronizing the Vision System and the Robot

To calibrate real-world measurements with the Vision System's world of pixels, it is necessary to perform the following steps:

1. Make sure that there is communication between the controller and the robot.
2. Click the ACL Terminal icon. Click the On-Line/Off-Line icon if necessary.
3. Execute the HOME routine.
4. Open Camera 1.
5. Check camera to make sure that the picture is adjusted, clear, and free of distortions.
6. Click the Calibration icon. This automatically establishes a connection between controller and robot.
7. Move the robot arm into desired position within the camera's field of vision.



- Using a pen or marker, mark a position on a piece of paper (or on the coordinate grid) close to the robot and strong enough to be seen in the camera picture. This position should be aligned with the TCP (tool center position – the exact center of the robot gripper). This first position is called Pos 1.



The screenshot shows the 'Calibration' window with 'Step 1' selected. Under 'Positions teaching', there are two tables for 'Pos1' and 'Pos2'. The 'Pos1' table has columns 1-5 and rows 0, X, Y, Z, P, R. The 'Pos2' table has the same structure. Below the tables are buttons for 'Record Position 1' and 'Record Position 2'. At the bottom, there is a section for 'Setting distance between positins [X-axis]' with a text box containing '0', a 'mm' unit, a 'Set distance' button, and 'Reset', 'Next Step', and 'Cancel' buttons.

	1	2	3	4	5
0	0	0	0	0	0
X	Y	Z	P	R	
	1690	0	5043	-636	0

	1	2	3	4	5
0	0	0	0	0	0
X	Y	Z	P	R	
	1690	0	5043	-636	0

- Click Record Position 1. (If using Controller-B, ensure that it is in Auto Mode.)

If trying to record a position produces a beep sound, the reason may be that the Teach Pendant has not been changed from Teach Mode to Auto Mode. (In the ACL Terminal type Auto, then click Enter.)

- Only along the X-axis, move the robot to a position on the X-axis that would be greater (+X) than Pos 1. Take a writing instrument and mark the position. This second position is called Pos 2.

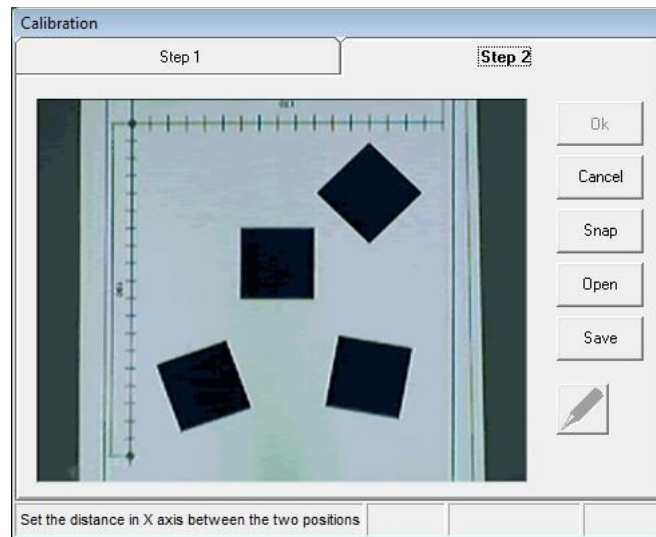
Click **Record Position 2**. (If using Controller-B, be sure it is in Auto Mode.)

- The distance between the two positions (Setting Distance) on the X-axis is displayed in Combo box t. To achieve the most accurate measurement, take a ruler and measure the distance between Pos 1 and Pos 2 by hand.

- Change the distance if necessary according to the hand measurement.

- Click **Set Distance**.

14. Click **Next Step**. This will take you to the Step 2 dialog box.



*Figure 24: Calibration Step 2 Dialog Box*

15. If necessary, move robot arm out of the camera's view.
16. Click **Snap** in the Step 2 dialog box.
17. You will be able to recognize Pos 1 and Pos 2 in the image.
18. Click the Pen icon. The Pen icon will start flashing after being engaged. Your mouse arrow will become a cross.
19. Click and drag from Pos 1 toward Pos 2. This movement represents an increase in the X-axis. (shows the direction of increase of the X-axis).
20. Click the Pen icon again. Now the X and Y-axes are set and the distances are in real-world millimeters.
21. When you move your mouse along the image, Pos 1 becomes coordinate 0,0 (center-point), and the directions of X and Y are now the same as the robot. The movement of the mouse along the axes shows changes in X and Y.
22. Click **OK**. The Vision System is calibrated and synchronized with the robot, and all new images will become calibrated.

### 3.4.2. Opening and Saving Calibration Files

The Configuration Step 2 dialog box enables file management operations.

#### **Save**

The Save icon saves the calibration. The extension of the files is .CAL.

## Open

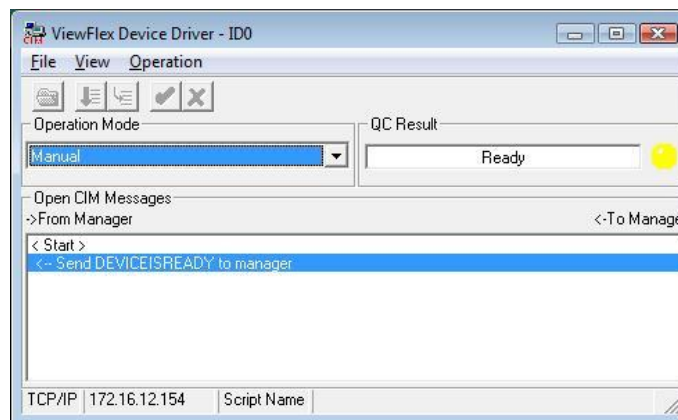
The Open icon will allow you to load backup calibration files. When the OK icon is clicked, the system is calibrated.

You can configure immediately in the Step 2 dialog box and open the calibrated file.

## 3.5. OPENCIM DEVICE DRIVER



The ViewFlex Device Driver interfaces between the OpenCIM network and the Vision Machine System as a quality control device.



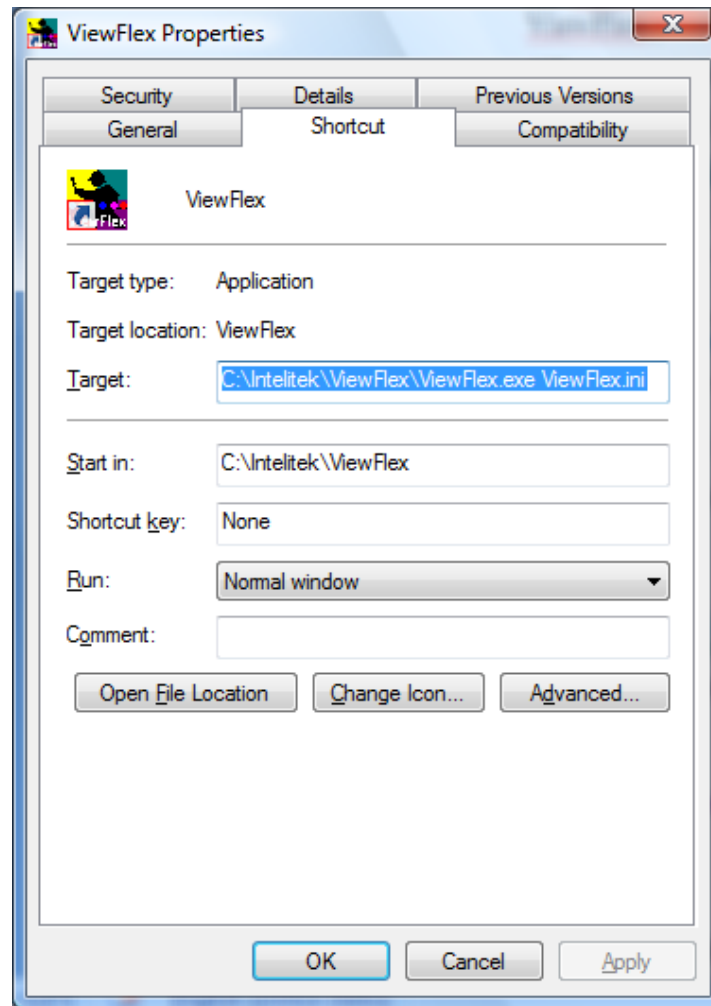
*Figure 25: ViewFlex Device Driver*

Each quality control test is defined as a separate process in the Machine Definition module. The quality control test consists of three parts:

- **File:** The File is the Script File (.bas) that contains the Program (Function) to be executed.
- **Program:** The Program is the Function from the File that returns the results of the quality control process to the OpenCIM Manager as Pass/Fail or Error.
- **Fail %:** Simulates test results (only in Simulation Mode) by determining the Pass/Fail according to the percentage of failure that was input.

The ViewFlex Device Driver performs the following functions:

- Activates a test on the Vision Machine System.
- Receives status messages from the Vision Machine System and from the OpenCIM Manager. (OpenCIM Messages dialog title in the ViewFlex.)
- Allows you to test and debug the Vision Machine process.
- Emulates the Vision Machine System in Simulation mode.



*Figure 26: ViewFlex Properties*

### 3.5.1. ViewFlex.ini

Be sure to update the script-path to the folder where the scripts files are. An example of where the directory that the script files are located in:

[Device Driver Definitions]

ScriptPath= C:\Users\Public\Documents\Intelitek\OpenCIM\Projects\LIB\viewflex

### 3.5.2. Operation Modes

The Operation Mode lets you define the control mode ViewFlex Device Driver is running in:

#### 3.5.2.1. On-Line

In On-Line mode the ViewFlex Device Driver waits for the commands from the OpenCIM Manager, and then Snap, Load Script, Execute Program Script, and Send Results.

### 3.5.2.2. Manual

Manual mode allows you to test the OpenCIM Manager and manually simulate conditions of Fail and Pass.

The following are activated only in Manual Mode, after a request from the OpenCIM Manager:



**Send Pass**

Sends pass to the manager.



**Send Fail**

Sends fail to the Manager.

#### **Simulation**

Simulation Mode allows you to simulate test results by determining the Pass/Fail according to the percentage of failure that was input to the OpenCIM Manager.

#### **Debug**

The Debug Mode allows you to debug every step of the script. This is made for testing the script before it will be run online.

The following are activated only in Debug mode:



**Open File** lets you load the script.



**Run Script** icon executes the script that has been chosen.



**Step-By-Step** icon lets you to execute each command line of the script individually.

### 3.5.3. Example of Script for Pass/Fail Test

The following script is an example for testing Pass/Fail Models. A Model refers to the pattern for which you are searching, and the image for which it is extracted.

The following program will try to find the “x” object by using the x-model (x.mod). If found, Fail is sent (QCR=“Fail”). If not found, then it will try to find the “v” object. If found, Pass is sent (QCR=“Pass”). If neither is found, Error is sent (QCR=“Error”).

For more information on the commands, refer to Inspector’s online help.

```
Function QCR() As String
Dim x As Integer
I_IMAGE1$ = Insptr.ImgGetCur
Insptr.ImgSetCurrent I_IMAGE1$, R_Def$, ALL_BANDS
' X.mod is a Model Search of object for Fail sign
M_X_MOD$ =
Insptr.PatLoad("C:\Users\Public\Documents\Intelitek\OpenCIM\Projects\MICROCIM\WS3\ViewFlex\X.mod")
Insptr.PatSetCur M_X_MOD$
Insptr.ImgConvertType(TO_8U)
Insptr.MeasNew()
x=Insptr.PatFind
If x=1 Then
QCR="Fail"
Else
' V.mod is a Model Search of object for Pass sign
M_Y_MOD$ =
Insptr.PatLoad("C:\Users\Public\Documents\Intelitek\OpenCIM\Projects\MICROCIM\WS3\ViewFlex\V.mod")
Insptr.PatSetCur M_Y_MOD$
x=Insptr.PatFind
If x=1 Then
QCR="Pass"
Else
QCR="Error"
End If
End If
Insptr.PatClose
Insptr.ImgClose
Insptr.MeasClose
Insptr.CloseAll
End Function
```