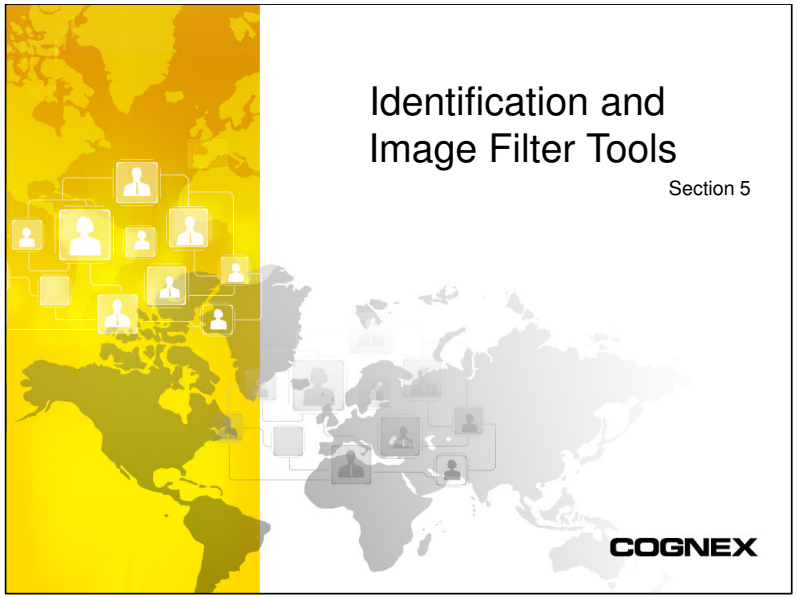
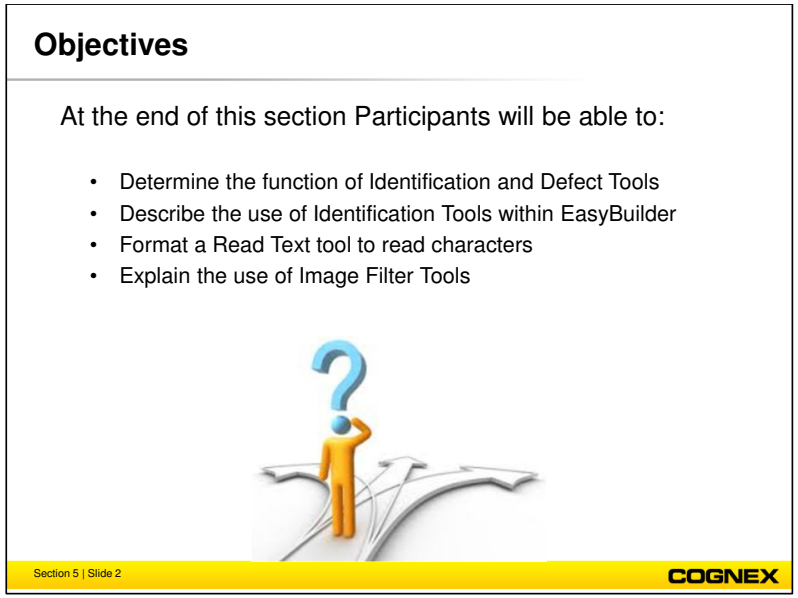


The fifth section of the In-Sight EasyBuilder Standard training will focus on **Identification and Filter Tools**.

Identification Tools are used to identify and verify 1D and 2D codes and symbols, alphanumeric text, pattern features and colors in the image.

The Image Filter Tools are used to enhance an image or area of the image for further image analysis.

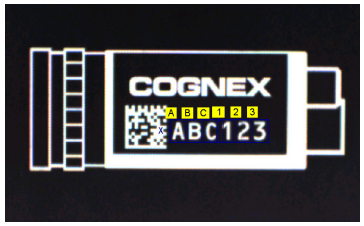


At the end of this section Participants will be able to:

- Determine the function of the Identification and Defect Tools
- Describe the use of Identification Tools within EasyBuilder
- Format a Read Text tool to read characters
- Explain the use of Image Filters

Inspect Part

Read a code



Read Text predefined or trained

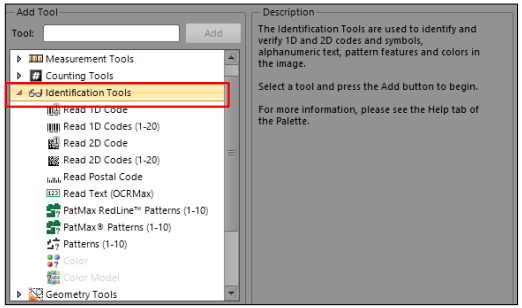
Section 5 | Slide 3 COGNEX

The **Inspect Part** step is used to assemble and configure the Inspection Tools that will be used to build your job. Examine your part or object to be inspected and determine which types of features best lend themselves to your application's requirements, and which Inspection Tools best suit themselves to solving your problem.

To add a tool to your job, select an appropriate tool from the Add Tool group, and press the **Add** button, or double-click on the tool. You will then be prompted to define the feature in the image, either through EasyBuilder Smart Feature selections, or by adjusting region graphics in the image or positioning geometric reference graphics.

After you have initially defined your feature, you can fine-tune the tool's controls by adjusting the various parameters.

Inspect Part – Identification Tools

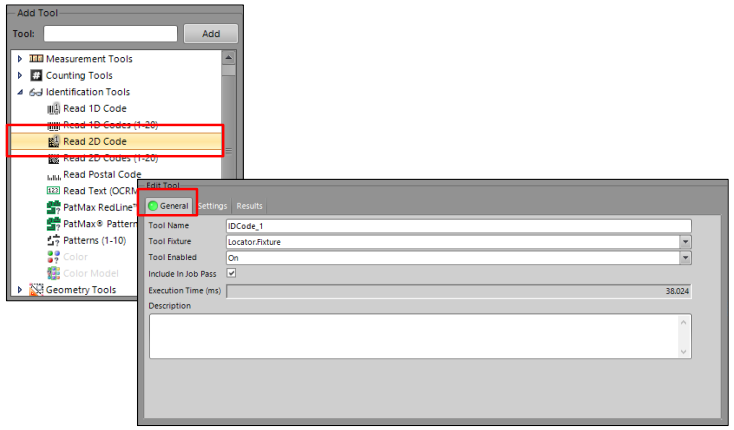


Section 5 | Slide 4 COGNEX

The **Identification Tools** are used to identify and verify 1D and 2D codes and symbols, alphanumeric text, pattern features and color in the image.

- The **Read 1D Code** and **Read 1D Codes (1 – 20)** Tools decode and/or verify ID and or Stacked bar codes, using the ReadIDMax® algorithm. The tools report the information that was decoded and a Pass if the code(s) were correctly decoded and the code was within the specified threshold limits.
- The **Read 2D Code** and **Read 2D Codes (1 – 20)** Tools decode and/or verify QR Code and/or Data Matrix symbologies, using the ReadIDMax® algorithm. The tools report the information that was decoded and a Pass if the code(s) were correctly decoded and the code was within the specified threshold limits.
- The **Read Postal Code** Tool decodes and/or verifies a single Posted Stacked Symbol within a region using the ReadIDMax® algorithm. The tool reports the information that was decoded and a Pass if the code/symbol was correctly decoded and the code/symbol was within the specified threshold limits.
- The **Read Text (OCR Max)** Tool is used to read and/or verify a text string within a region, after training and creating user-defined character fonts.
- The **PatMax® Patterns (1 – 10)** Tool is used to train occurrences of pattern features, using PatMax algorithms, and then verify or identify the particular occurrence of the pattern features in the image.
- The **Patterns** Tool is used to train occurrences of pattern features, and then verify or identify the particular occurrence of the pattern features in the image.
- The **Color** Tool is used to determine, from a library of trained Colors, which Color best matches the color in the image.
- The **Color Model** Tool is used to determine, from a library of Color Models, which Color Model best matches the colors in the image, returning the name of the matching Color Model.

Read 2D Code Tool



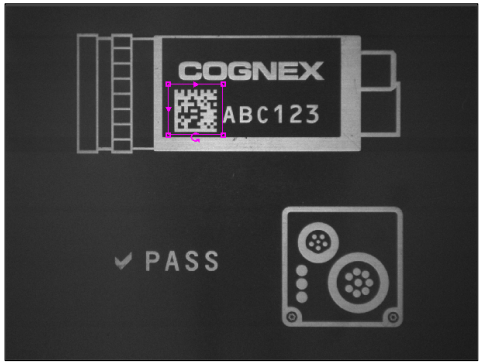
Section 5 | Slide 5 COGNEX

The **Read 2D Code** Tool reads and/or verifies information contained in a single 2D code, using the ReadIDMax® algorithm. It reports the decoded information and a Pass code if the code is correctly decoded within the specified limits, a Fail if it cannot be decoded or it's outside limits.

Click the **Add** button to begin.

NOTE: *Cognex recommends that the function be trained when the 2D symbols used in the application all share the same characteristics. If the 2D symbols can vary in grid size or polarity, for example, do not train the function and allow it to consider all possible settings when decoding a 2D symbol.*

Position Region

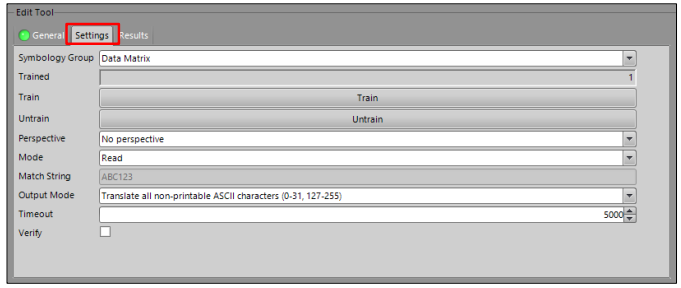


Section 5 | Slide 6 COGNEX

Position the region by dragging the magenta region to the area of the image that you would like to enhance. Once you have configured the search region, click the **OK** button.
In this slide we our region is the 2D code.

Read 2D Code – Parameters

Settings Tab



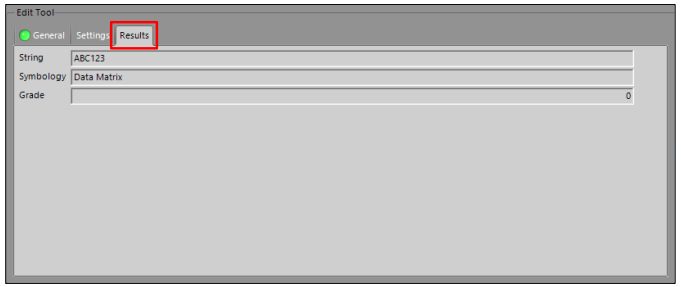
Section 5 | Slide 7 COGNEX

The Read 2D Code Tool's **Settings Tab** contains the following information:

- *Symbology Group* – Defines the type of symbology (Data Matrix or QR Code; default = Data Matrix) to attempt to decode.
  NOTE: *When QR Code is selected, the tool will attempt to decode all QR Code types (QR Model ½, Micro QR and QR Model ½).*
- *Trained* – Displays whether the tool has been trained on a particular code/symbol; displays 1 when the tool has been trained, and a 0 when it has not been trained.
- *Train* – Trains a model of the code/symbol read; trained information is retained until the Untrain button is pressed. Training is used to help the speed and performance of the tool, by teaching the tool which type of code/symbol to decode.
- *Untrain* – undoes the training of a trained model code/symbol. To retrain a symbol, press the Untrain button, then press the Train button with the new code present.
- *Perspective* – Defines the decoding settings (No Perspective, Perspective or Train Perspective; default = No Perspective) for Data Matrix symbologies. The Symbology Group parameter must be set to Data Matrix to enable this parameter.
- *Mode* - Defines the operational mode (Read or Match String; default = Read) of the tool.
- *Match String* – Defines the expected string to be read (only enabled when the Mode parameter is set to Match String). Enter the information that you expect the tool to decode.
- *Output Mode* – Defines how to process non-printable ASCII characters.
- *Timeout* – Defines the amount of time, in milliseconds (0 – 30000; default = 0), that the tool will search for a valid code/symbol before execution is stopped and the tool results in a Fail.
- *Verify* – Defines whether the print quality tests will be performed (by default, this setting is disabled).

Read 2D Code – Parameters

Results Tab

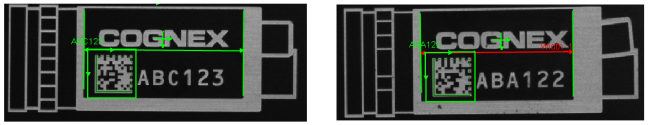


Section 5 | Slide 8 COGNEX

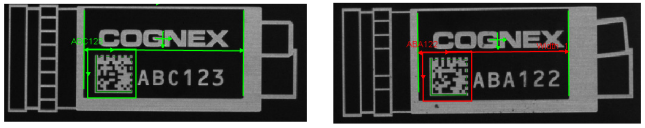
The **Results** Tab displays the following information about the decoded code/symbol:

- *String* – Returns the string contained in the decoded code/symbol.
- *Symbology* – Returns the type of symbology that was decoded.
- *Grade* – Returns the grade given to the decoded code/symbol.

Results



Mode set to *Read*



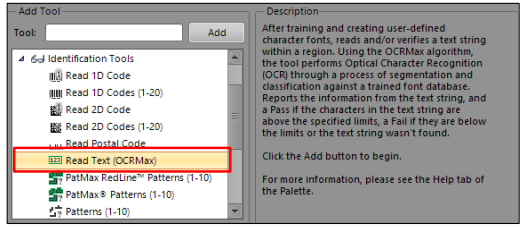
Mode set to *Match String*

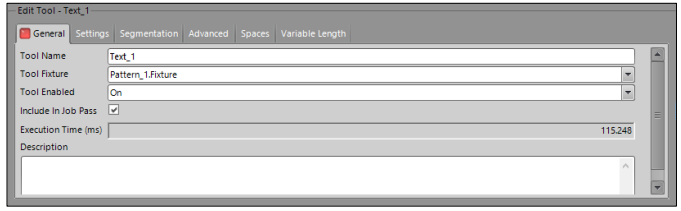
Section 5 | Slide 9 COGNEX

The Mode in the top image was set to **Read**. This tells the tool to simply read the letters, as a result letters on the second example (ABA122) match letters that were trained in the first example (ABC123). Since there are no new letters both examples are showing as a Pass.

The Mode in the bottom image was set to **Match String**. The tool is looking for an exact match to what was entered in the Match String field in the Settings tab. The tool is looking for ABC123 in both examples, since it is not an exact match in the second example, the job will Fail.

Read Text (OCRMax)



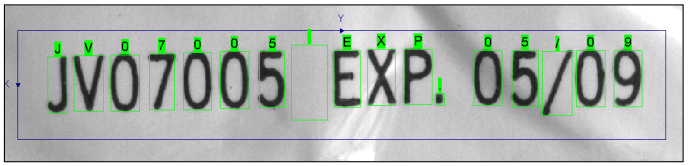


Section 5 | Slide 10 COGNEX

The **Read Text** Identification Tool is used to read and/or verify alpha-numeric text strings, using trained character models.

Using the OCRMax algorithm, the tool performs Optical Character Recognition (OCR) through the process of segmentation and classification against a trained font database. It reports the information from the text string and will return a Pass if the characters in the text string are above the specified limits, and a Fail if they are below the limits or the text string is not found.

OCR and OCV



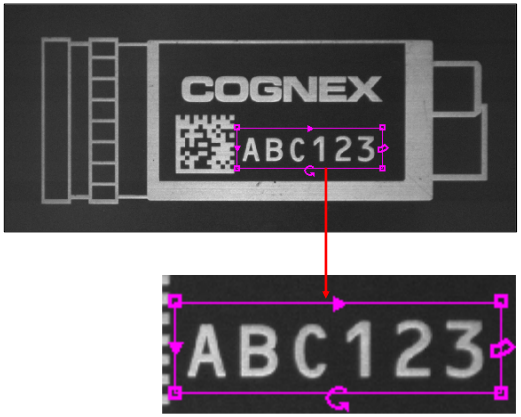
OCR = Optical Character Recognition
OCV = Optical Character Verification

Section 5 | Slide 11 COGNEX

Optical Character Recognition (OCR) is the process of reading unknown characters and pattern matching the results against character models to 'read' characters and pass them along as a string. When recognizing (reading) the tool compares each character against all trained characters in the database of trained characters for a match.

Optical Character Verification (OCV) is the process of reading known characters and pattern-matching the results against character models to 'verify' that an expected string was present. When verifying the tool compares each character only against the character designated as its identity for a match.

Position Region



Section 5 | Slide 12 COGNEX

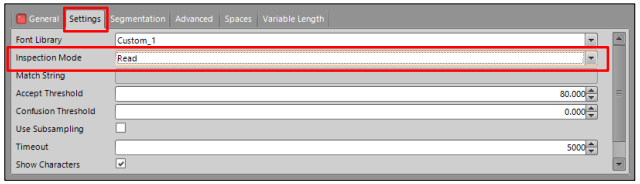
Position the region over the text string that you want to read or verify. The line of text within the region is split into images of the individual characters, and each character is enclosed within a non-editable rectangle. The region defines the approximate location, angle and skew of the line of text.

Once the region has been set, click the **OK** button.

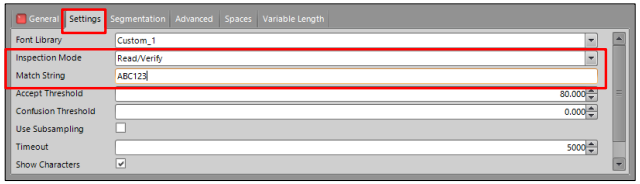
Read Text

Settings Tab – Inspection Mode

Read



Read/Verify



Section 5 | Slide 13

COGNEX

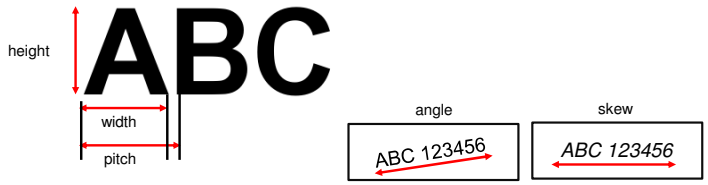
The **Inspection Mode** specifies the inspection mode of the function during run-time.

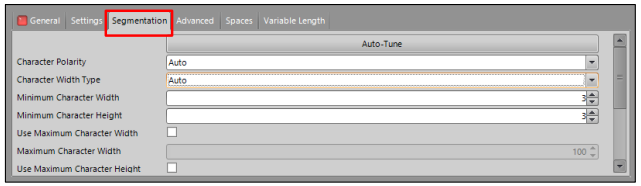
- *Read* is the default – the function will attempt to read the characters in the ROI, based on the trained instances of the characters.
- *Read/Verify* – the function will attempt to first read the characters in the ROI, based on the trained instances of the characters, and then verify that the read characters correspond to the text string specified in the Match String field.

- *Match String* – specifies the text string that must be correctly matched when the Inspection Mode is set to Read/Verify.

Read Text

Segmentation Tab





Section 5 | Slide 14

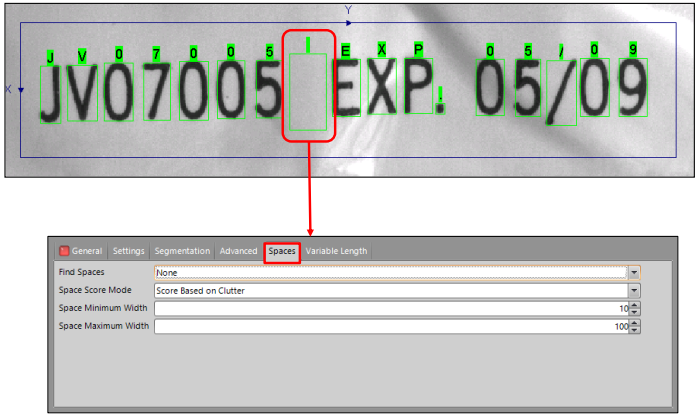
COGNEX

During the **Segmentation** process, the OCRMax function determines the location of the line of text within the ROI, and calculates the text's angle, skew and polarity. The region is then normalized to remove unwanted noise before being binarized into foreground and background pixels. Within the binarized image, blob analysis is performed to produce character fragments, with each character fragment representing a single blob. The character fragments are then grouped together to form characters, and the characters are assigned a character region. The character region is a tight, non-editable bounding box enclosing all of the foreground (i.e. ink) pixels in the ROI.

The line of text within the ROI is split into images of the individual characters, and each character is enclosed within a non-editable character rectangle. The ROI defines the approximate location, angle and skew of the line of text. The Angle Range and Skew Range parameters on the Segmentation tab can be used to compensate for variations, if necessary.

Read Text

Spaces Tab



Section 5 | Slide 15 COGNEX

The **Spaces** tab specifies how the function will account for spaces between characters if necessary. The OCRMax function processes inter-character gaps by classifying then as space characters, and these are user defined.

Find Spaces specifies how the function will handle the insertion of space characters into gaps between other characters.

None is the default – the function will never insert a space character, regardless of how large the inter-character gap between characters.

Insert Single Space – the function will insert a single spaces character per inter-character gap, regardless of how large the gap between characters.

Insert Multiple Spaces – the function will insert x-number of space characters per inter-character gap.

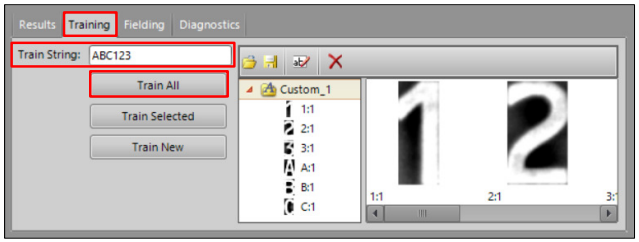
Space Score Mode specifies how the function will calculate scores for space characters.

Always Score 100 is the default, the space characters will always receive a score of 100.

Score Based on Clutter – the score of a space character is based on the fraction of pixels that are background; a space character that consists entirely of background pixels will receive a score of 100.

Read Text

Training Tab



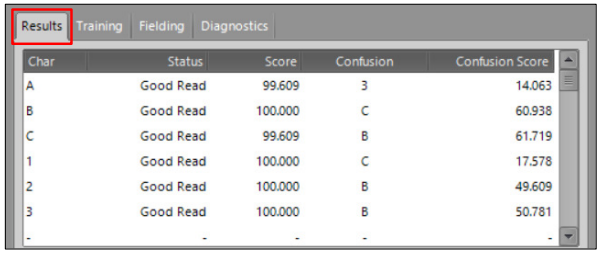
Section 5 | Slide 16 COGNEX

The **Training** tab is used to train, view, rename and remove characters. The tab is divided into two groups, Training, where the character training parameters are defined, and Characters, where the trained fonts are managed.

- *Train String* specifies the text string to be trained. The number of characters in the Train String text entry box must match the number of segmented characters. A maximum of 50 characters can be trained.
- *Train All* specifies that all of the characters in the region will be trained. Enter the characters to be trained in the Train String text entry box before clicking the Train All button. The number of characters in the Train String text box must match the number of segmented characters.
- *Train Selected* specifies that specific characters in the region will be trained. When this option is selected, the Train String entry box will be disabled, and the Train Selected button will launch the Add Selected Characters to Font dialog. This dialog contains an unwrapped image of the region, with a label and text-entry boxes below each of the segmented characters. The label is the currently associated symbol for that character (a "?" denotes an unknown or untrained character). Below the label is a text-entry box. Enter a label for each segmented character in the text-entry box; leave the text-entry box empty to not retrain characters. Press the **Train** button to close the dialog.
- Train New specifies that only new characters in the region will be trained. The characters expected to be trained are entered in the Train String entry box, before the Train button is clicked. The number of characters in the Train String entry box must match the number of segmented characters.

Read Text

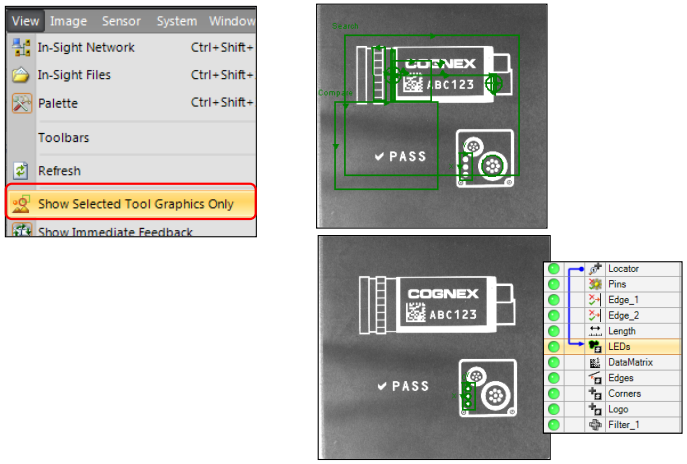
Results Tab



The **Results** tab displays the OCRMax function's results.

- *Char* displays the character read at that position.
- *Status* displays the status (Bad Read, Good Read, Ignored, Confused, Mismatch, Confused Mismatch, No Read, or Validation Failed) of the character.
- *Score* displays a score that measures how closely the read character matched the trained character in the font database.
- *Confusion* displays the next, best matching character.
- *Confusion Score* displays a measure of how certain the function is of the chosen match, when compared against the next, best match.

Showing Graphics Tool by Tool



When you have many tools in the Results list, it can be difficult to determine which tool corresponds to which region on the image. The default is to show all tools.

You can change this with:

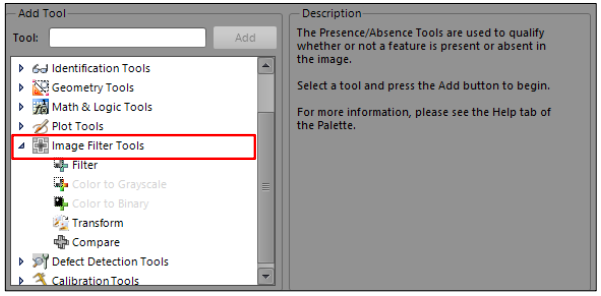
View → Show Selected Tool Graphics Only

This will display the region for the individual tool you select.

This menu item is a toggle. You can return to showing all graphics by clicking again on:

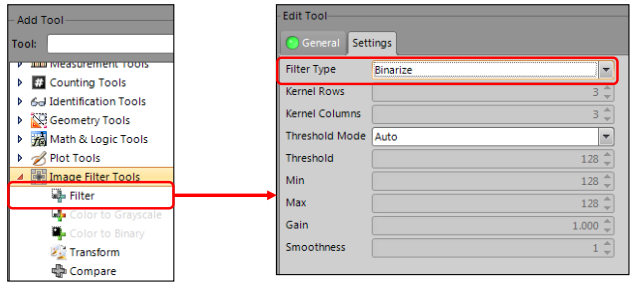
View → Show Selected Tool Graphics Only

Inspect Part – Image Filter Tools



The **Image Filter Tools** are used to enhance an image or area of the image for further image analysis.

Inspect Part – Filters

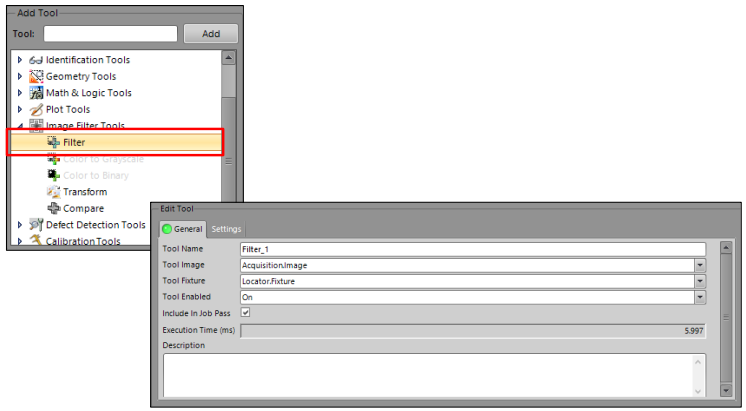


The Filter Tool's **Settings Tab** contains the following information:

Filter Type – Defines the image enhancement technique that will be applied to the region. The result of the image enhancement technique will automatically update within the region.

- *Binarize* (default) – Filters the image based on a greyscale threshold into white (255) or black (0); outputs an image where input pixel values equal to or above the threshold value are white, and values below the threshold are black.
- *Clip* – Filters the image by comparing the grayscale value of each individual pixel to a minimum and maximum greyscale value; outputs an image where the extreme ends of the greyscale spectrum are eliminated so features are more uniform.
- *Equalize* – Filters based on a histogram equalization, where the most frequent greyscale values are equalized to create a more equal distribution; outputs an image with increased contrast.
- *Greyscale Distance* – Filters based on the greyscale value of each output pixel as the difference between the greyscale value of each input pixel and the threshold level; output an image where, for example, if the input pixel value is 100 and the threshold is 25, the output pixel value is 75.
- *Invert* – Filters based on a greyscale inversion of pixel values; outputs an image that is a 'negative' of the input image.
- *Threshold Range* – Filters based on a binary (black and white) threshold which compares each input pixel with two threshold values (minimum and maximum) to determine whether the output pixel is white or black; outputs an image where if the input pixel value is equal to or exceeds the minimum or maximum, the output pixel is white, and if the input pixel value is within the minimum or maximum, the output pixel is black.
- *Optical Density* - Filters based on an inversion of pixel values on a logarithmic scale; outputs an image that reveals the density of objects and features by measuring the amount of light passing through them. Denser objects and features are represented by lighter pixels in the output image.
- *Stretch* - Filters by linearly scaling or "stretching" the greyscale values; outputs an image with increased contrast.

Inspect Part – Filter Image Tool



The **Filter Image Filter Tool** filters a region of the image with both local and global image-enhancement techniques, which are defined by selecting a Filter Type; outputs a Tool Image that can be referenced by other Location or Inspection Tools.

This tool is a multipurpose tool and produces a color or greyscale output image that is used to enhance low contrast images and/or expose features in an image. This tool helps to accentuate the desired features, and minimizes the undesired features, for most parts. When used effectively, this tools makes inspection applications more reliable and repeatable.

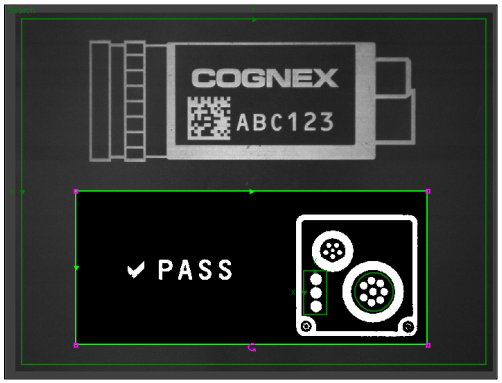
Inspect Part – Filter Image Tool

Settings Tab

The Image Filter Tool **Settings Tab** includes the following parameters:

- *Filter Type* – defines the image enhancement techniques that will be applied to the region. The result of the image enhancement technique will automatically update within the region.
- *Kernel Rows* – specifies the number of rows (1-25; default = 3) of the kernel. A kernel is a rectangular array of numbers that defines the processing neighborhood for operations performed on each pixel location of an image.
- *Kernel Columns* – specifies the number of rows (1-25; default = 3) of the kernel.
- *Threshold Mode* – specifies whether the tool will automatically determine a threshold level when the Filter Type is set to Binarize, Greyscale Distance or Threshold Range.
- *Threshold* – specifies the threshold levels (0-255; default = 128) when the Filter Type is set to Binarize, Greyscale Distance or Threshold Range.
- *Min* – specifies the minimum greyscale value (0-255; default = 128) when the Filter Type is set to Clip, Stretch, or Threshold Range.
- *Max* – specifies the maximum greyscale value (0-255; default = 128) when the Filter Type is set to Clip, Stretch, or Threshold Range.
- *Gain* – specifies a gain value 9 0 to 10; default = 1) when the Filter Type is set to Sharpen. This parameter dictates the strength of the sharpening that will be performed by controlling the amount of edge contrast enhancement.
- *Smoothness* – specifies the smoothness value (1 to 4; default = 1) for the Gaussian kernel when the Filter Type is set to Sharpen. Smoothness controls the standard deviation of the Gaussian kernel' higher smoothness values result in a wider aperture for the smoothing operation helping to sharpen features.

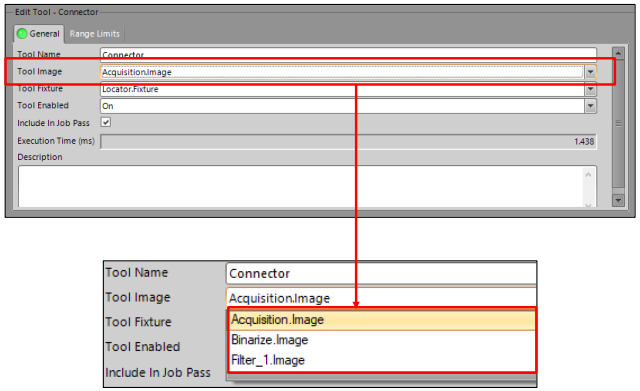
Point Filter – Binarize



Section 5 | Slide 23 COGNEX

Binarize filters the image based on a greyscale threshold into white (255) or black (0); outputs an image where input pixel values equal to or above the threshold value are white, and values below the threshold are black.

Using a Filtered Image

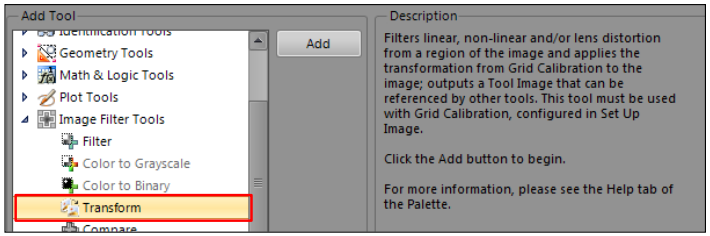


Section 5 | Slide 24 COGNEX

Once you have created a filtered image, EasyBuilder will allow you to select it for use in any vison tool.

Grid Calibration – Transform

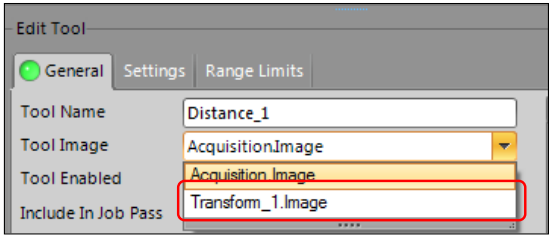
In order to “undistort” the image, you need to add a Transform Filter



The **Grid Calibration** process results in conversion data for each distance (point-to-point or square to square) being stored in the job. Now you need to apply this information to the original image, to yield a new image (**Transform**) that is corrected for lens distortion and that is in real-world units.

Grid Calibration – Transform

- Select the **Transform Image** in the General tab of the measurement tool

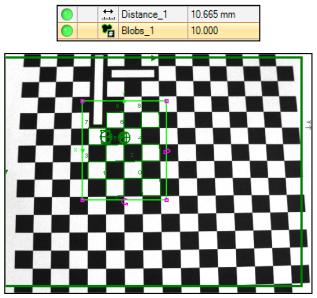


- The measurement will now be based on the undistorted image and be in real-world units

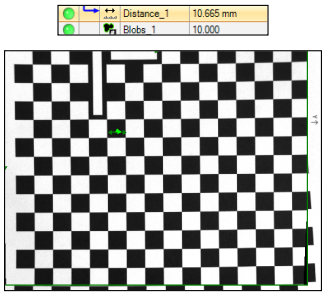
Once you add a Transform filter to the job, all measurement tools will allow you to choose the transformed image in the **Tool Image** parameter in the General tab of the tools.

Grid Calibration – Viewing Undistorted Image

You will see either the original image or the transformed image, depending on which the selected tool uses



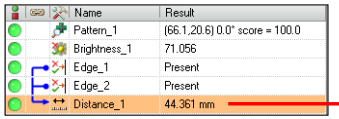
Blobs tool uses A0.
Displayed image is distorted.



Distance tool uses Transform image.
Displayed image is undistorted.

COGNEX

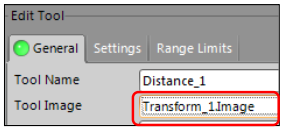
Results: Decreasing Distortion

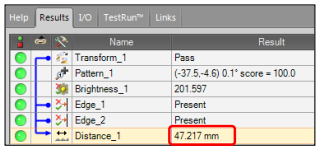


Results are based on original image, which may have distortion from lens

To decrease distortion:

1. Use Grid Calibration
2. Add **Transform** tool
3. Have measurement tool reference **Transform** in General tab:





COGNEX

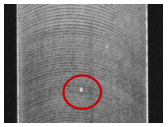
This slide displays the **Results** based upon the distance parameters that were entered for this job.

- The first image has a distance that falls within the set limits, so this job is a Pass.
- The second image has a distance that falls outside of the set limits, so this job is a Fail.

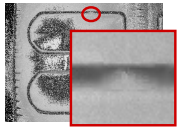
Applications

Which filter(s) did you use?

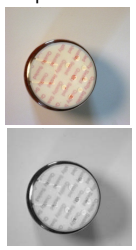
Wood Grain



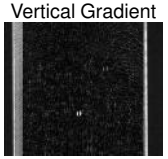
Blister Pack



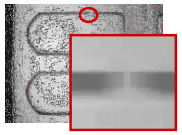
Spice Jar



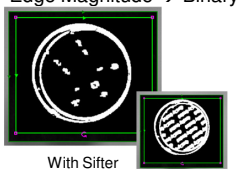
Vertical Gradient



Close



Edge Magnitude → Binary



With Sifter

Without Sifter

COGNEX

Which filter(s) would you use in each example?

- Wood Grain
- Blister Pack
- Spice Jar
- Vertical Gradient
- Close
- Edge Magnitude

Summary

- There are many ways to identify parts, such as Read 2D Code & OCRMax Tools
- ID tools have two modes:
  - Read (or Identify)
  - Verify (or Match)
- The Read Text tool can be set up to Read or Verify a string of text
- Image Tools help vision tools by enhancing desired features and minimizing other undesired ones

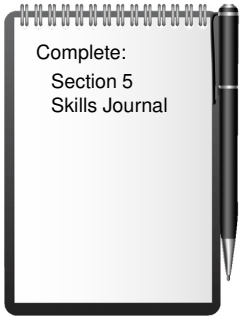
COGNEX

In this section we covered the following topics:

- There are many ways to identify parts, such as Read 2D Code & PatMax Patterns
- ID tools have two modes:
  - Read (or Identify)
  - Verify (or Match)
- The Read Text tool can be set up to Read or Verify a string of text
- Image Tools help vision tools by enhancing desired features and minimizing other undesired ones

Skills Journal

Lab Exercise



Section 5 | Slide 31

COGNEX

Complete:

Skills Journal (image designed by pngtree)

Lab Exercise