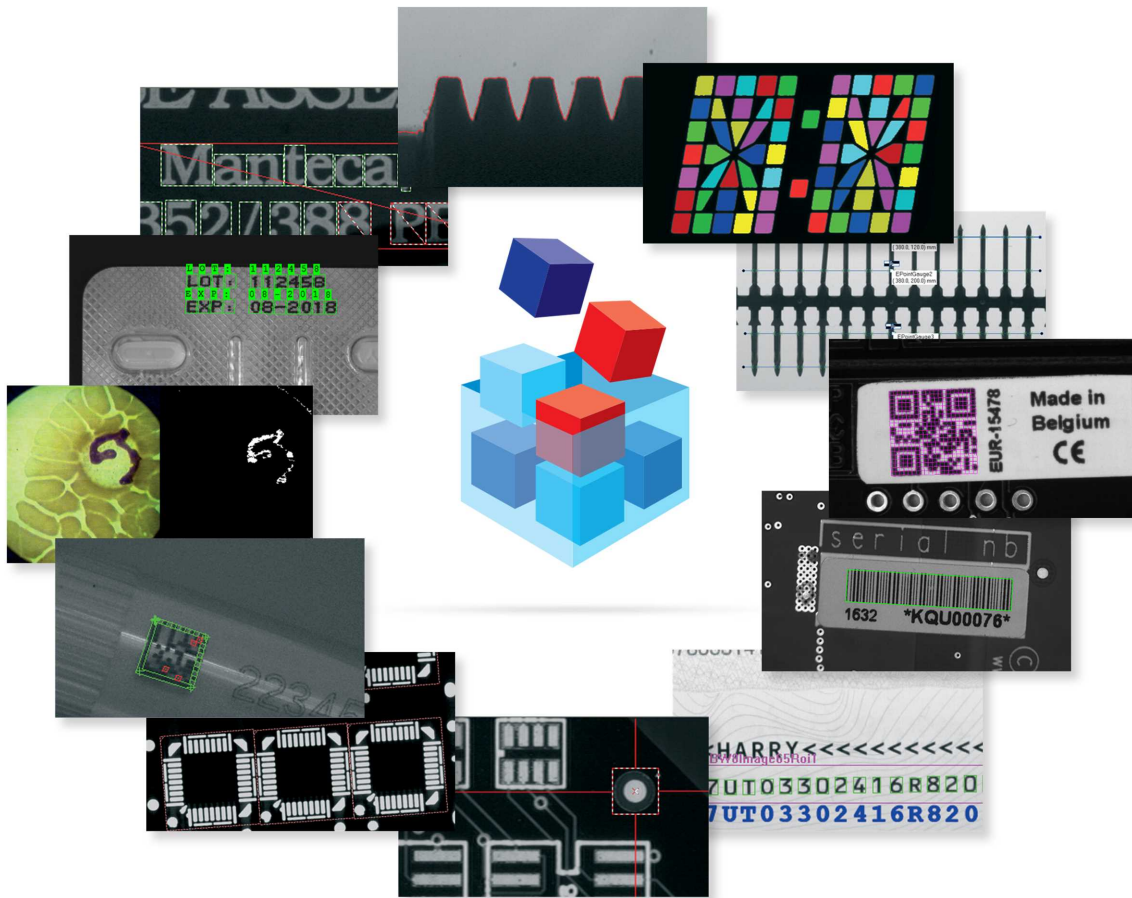




사용자 설명서

Open eVision Studio



사용 약관

EURESYS s.a. 는 EURESYS s.a.의 하드웨어 및 소프트웨어의 부속 문서, 상표의 모든 재산권, 소유권, 이권을 보유합니다.

이 설명서에 언급된 회사 및 제품의 모든 이름은 해당 소유자의 상표일 수 있습니다.

이 문서에 포함된 EURESYS s.a.의 자료, 하드웨어 또는 소프트웨어, 브랜드를 사전 통지 없이 라이선싱, 사용, 임대, 임차, 번역, 재현, 복사 또는 수정하는 행위는 허용되지 않습니다.

EURESYS s.a. 는 언제든지 자사 재량에 따라 사전 통지 없이 제품 사양을 수정하거나 이 문서에서 제공하는 정보를 변경할 수 있습니다.

EURESYS s.a. 는 EURESYS s.a.의 하드웨어 또는 소프트웨어 사용과 관련하여 발생하는 일체의 매출, 수익, 영업권, 데이터, 정보 시스템의 손실 또는 피해 또는 기타 특별하거나, 우발적이거나, 간접적이거나, 필연적인 또는 징벌적인 손해에 대해 책임을 지지 않으며, 이는 본 문서의 누락 또는 오류로 인한 결과일 경우에도 마찬가지입니다.

이 문서는 Open eVision Studio 2.7.1의 부속 자료입니다(문서 빌드 1114).
© 0000 EURESYS s.a.

목차

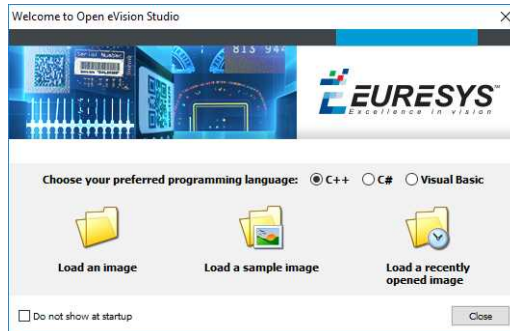
1. Open eVision Studio 사용하기	6
1.1. 프로그래밍 언어를 선택하십시오.	7
1.2. 인터페이스 둘러보기	8
1.3. 이미지에서 도구를 실행하기	9
1단계 : 도구 구성하기	9
2단계 : 이미지 열기	10
3단계 : ROI 관리하기	10
4 단계 : 도구 구성하기	12
5 단계 : 도구 실행 및 실행 시간 확인하기	13
6단계: 생성된 코드 사용하기	15
1.4. 이미지 전처리 및 저장	16
2. Tutorials	18
2.1. Preliminaries	18
2.2. EasyImage	18
Converting a Gray-Level Image into a Binary Image	18
Extracting an Object Contour	20
Transforming a Gray-Level image into its Black and White Edges	21
Detecting the Corners of an Object Using Harris Corner Detector	22
Detecting a Horizontal or Vertical Line Using Projection	23
Creating a Flexible Mask	24
Computing Gray-Level Statistics Using a Flexible Mask	25
Detecting the Corners of an Object Using Hit-and-Miss Transform	26
Extracting a Vector Using Profile Function	27
Enhancing an X-ray image	28
Correcting Non-Uniform Illumination	29
Correcting Shear Effect	30
Correcting Skew Effect	32
2.3. EasyColor	33
Performing Thresholding on Color Images	33
Performing Color Segmentation	34
2.4. EasyObject	35
Removing Non-Significant Objects After Image Segmentation	35
Detecting Differences Between Images Using Min-Max References	37
Detecting Printing Errors Using a Flexible Mask	39
2.5. EasyMatch	41
Learning a Pattern and Creating an EasyMatch Model File	41
Matching a Pattern According to a Model File	41
Learning a Pattern According to an ROI	42
Improving the Score of Matching Instances by Using "Don't Care Areas"	44
2.6. EasyFind	46
Detecting Highly-Degraded Occurrences of a Reference Model in Multiple Files	46
Improving the Score of Found Instances by Using "Don't Care Areas"	48
2.7. EasyGauge	50
Measuring the Rotation Angle of an Object	50
Measuring the Diameter of a Circle	52

Measuring a Distorted Rectangle	53
Locating Points Regarding to a Coordinate System	55
Unwarping a Distorted Image	57
2.8. EasyOCR	59
Learning Characters and Creating an EasyOCR Font	59
Recognizing Characters According to a Font	61
2.9. EasyOCV	62
Creating an EasyOCV Model File	62
Inspecting Characters in an Image According to a Model File	63
Inspecting Characters in an ROI According to a Model File	64
Learning a Model Using Statistics (1)	66
Learning a Model Using Statistics (2)	69
2.10. EasyBarCode	70
Reading Bar Codes Automatically	70
2.11. EasyMatrixCode	72
Reading Data Matrix Codes Automatically	72
Learning a Data Matrix Code and Creating an EasyMatrixCode Model File	73

1. Open eVision Studio 사용하기

1.1. 프로그래밍 언어를 선택하십시오.

처음 시작할 때 Open eVision Studio 다음과 같은 웰컴스크린을 표시합니다.



1. 프로그래밍 언어를 선택하십시오.

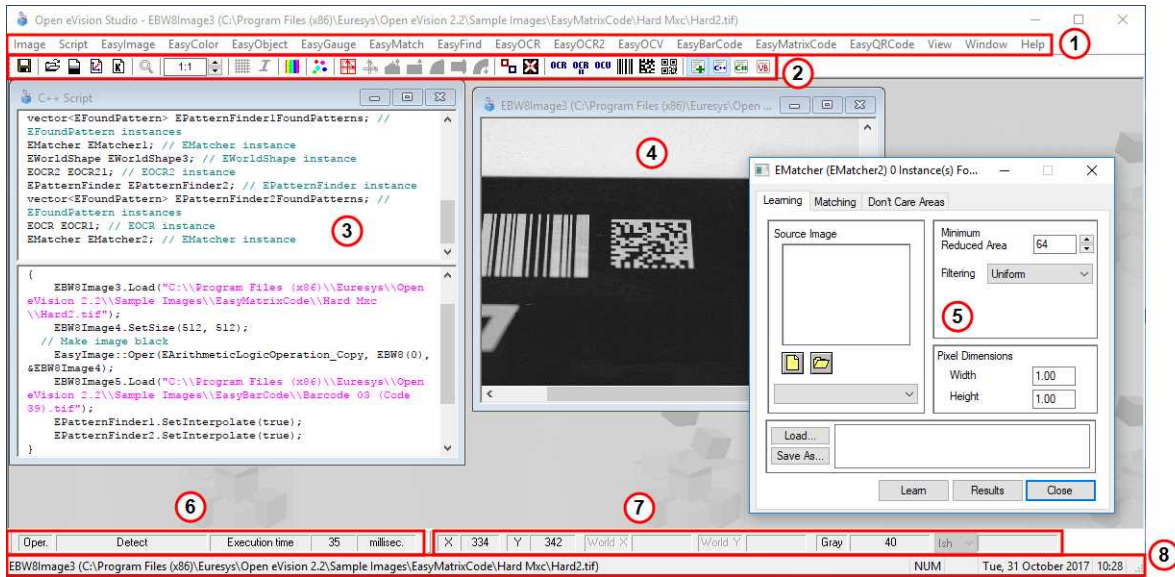
선택한 것을 저장하고 다음 시작 시에는 사용자의 프로그래밍 언어를 자동으로 선택합니다
Open eVision Studio.

참고: 프로그래밍 언어를 바꾸면 스크립트 윈도우에 있는 어떤 스크립트든지 자동으로 삭제되고 윈도우 내용이 초기화됩니다.

2. 이미 불러들인 한 개 혹은 몇 개의 이미지를 나중에 처리하기 위해서 불러오기 버튼 중 하나를 클릭하십시오.
3. 다음 시작시 웰컴 스크린을 보고 싶지 않으면 시작할 때 보지 않기 박스를 체크하십시오.
Open eVision Studio.

웰컴 스크린을 보려면 언제든지 **도움 > 웰컴 스크린 메뉴**로 가서 설정을 변경하십시오.

1.2. 인터페이스 둘러보기



Open eVision Studio 그래픽 유저 인터페이스 (GUI) 는 다음과 같이 이루어져 있습니다:

1. 메인 메뉴 바에서 모든 라이브러리의 기능과 툴을 이용할 수 있습니다.

Open eVision Studio 라이선스가 필요하지 않으며 모든 라이브러리를 시험해볼 수 있습니다. 물론 Open eVision Studio 에서 사용자의 애플리케이션에 코드를 복사하였지만 필요한 라이선스가 없으면 실행시 “라이선스 없음” 오류 메시지를 보여줍니다.

2. 주요 도구 바는 이미지, 모양, 게이지, 바코드, 매트릭스 코드와 같은 주요 Open eVision 개체에 간편하게 액세스 할 수 있습니다.

3. 스크립트 윈도우는 Open eVision Studio에서 수행한 작업에 해당하는 코드를 선택한 프로그래밍 언어로 표시합니다. 언제든지 이 코드를 자신의 응용 프로그램에 저장하거나 복사할 수 있습니다.

4. 이미지 윈도우에는 라이브러리 및 도구를 사용하여 처리 할 수 있는 오픈 이미지가 표시 됩니다.

5. 도구 윈도우를 사용하여 사용 가능한 모든 도구를 쉽게 구성할 수 있습니다. 해당 설정이 스크립트 윈도우에 자동으로 추가되어 쉽게 재사용할 수 있습니다.

대부분의 도구 창은 유동적이어서 화면 크기를 보다 잘 활용하기 위해 Open eVision Studio 기본 창 밖으로 쉽게 이동할 수 있습니다.

6. 실행 시간 막대는 컴퓨터에서 선택한 기능 (밀리 초 또는 마이크로 초 단위로 측정) 실행에 소요된 정확한 시간을 표시합니다. 이 정확한 측정은 애플리케이션의 성능을 평가하는데 도움이 됩니다.

7. 색상 도구 모음에는 이미지 커서의 X 및 Y 좌표와 해당 픽셀 값과 같은 현재 정보가 표시 됩니다.

8. 상태 표시 줄에는 활성 이미지 파일 경로와 같은 애플리케이션에 대한 일반 정보가 표시 됩니다.

1.3. 이미지에서 도구를 실행하기

1단계 : 도구 구성하기

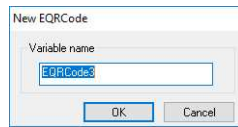
일반적으로 첫 번째 단계는, Open eVision Studio을 사용할 때, 이미지에 사용할 라이브러리와 도구를 선택합니다.

그렇게 하려면 :

1. 기본 메뉴 표시 줄에서 사용할 라이브러리를 클릭하십시오.
2. 사용할 도구를 클릭하십시오.

모든 라이브러리 (EasyImage, EasyColor 및 EasyGauge 제외)는 **New Xxx Tool**이라는 하나의 도구만 표시합니다. 이 라이브러리 중 일부는 추가 기능도 제공합니다.

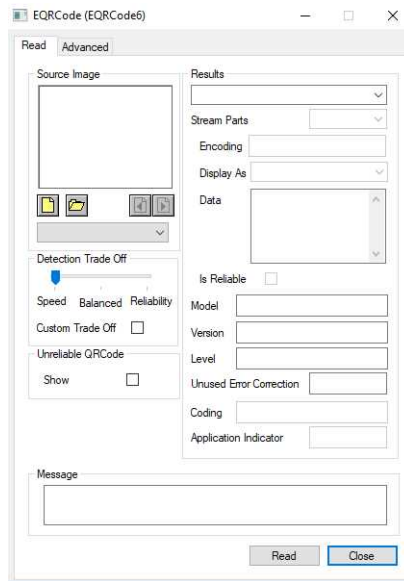
3. 대화 상자에서 자동으로 생성되고 처리 결과가 포함될 변수의 **Variable name** 을 입력하십시오.



EasyQRCode에 대한 변수 생성 대화 상자의 예

4. Ok를 클릭하십시오.

선택한 도구 대화 상자가 열립니다.



EasyQRCode에 대한 변수 생성 대화 상자의 예


다음 단계는 "2단계 : 이미지 열기" 아래.

2단계 : 이미지 열기

라이브러리와 도구를 선택했으면 이 도구를 적용할 이미지를 열어야 합니다.

선택한 도구 대화 상자의 소스 이미지 영역에서 다음을 수행합니다.

1. 이미지 열기:

- Click on the  이미지 열기 버튼을 클릭하고 컴퓨터에서 하나 이상의 이미지 (SHIFT 및 CTRL 사용)를 선택하십시오.
- 또는 이미 드롭 다운 목록에서 열려있는 이미지 중 하나 (또는 ROI가 있는 경우) 중 하나를 선택하십시오.

참고: 적절한 파일 형식 (JPG, PNG, TIFF 또는 BMP) 및 라이브러리에 따라 8비트 및 / 또는 24 비트의 이미지만 선택할 수 있습니다.



- ### 2. 이미지를 여러 개 선택한 경우 Load Previous 또는 Load Next 버튼으로 이미지를 활성화하십시오.

이 도구는 로드된 이미지에 자동으로 적용되며, 이 단계에서 도구 기본 설정에 따라 결과가 표시됩니다.

다음 단계는 "3단계 : ROI 관리하기" 아래.

3단계 : ROI 관리하기

몇 가지 경우에, 대부분 처리 시간을 줄이거나 읽으려는 대상을 단일화하기 위해서는, 전체 이미지를 처리하는 것이 아니라 이미지의 하나 또는 여러 개의 잘 정의된 사각형 부분 또는 ROI (관심 영역)를 처리하기를 원하지 않을 것입니다.

Open eVision에서 ROI는 이미지에 첨부되어 있으며 부모 이미지가 있는 동안에만 존재합니다.

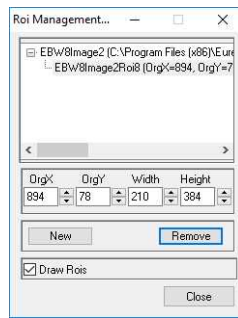
ROI 생성하기

1. 이미지 열기:

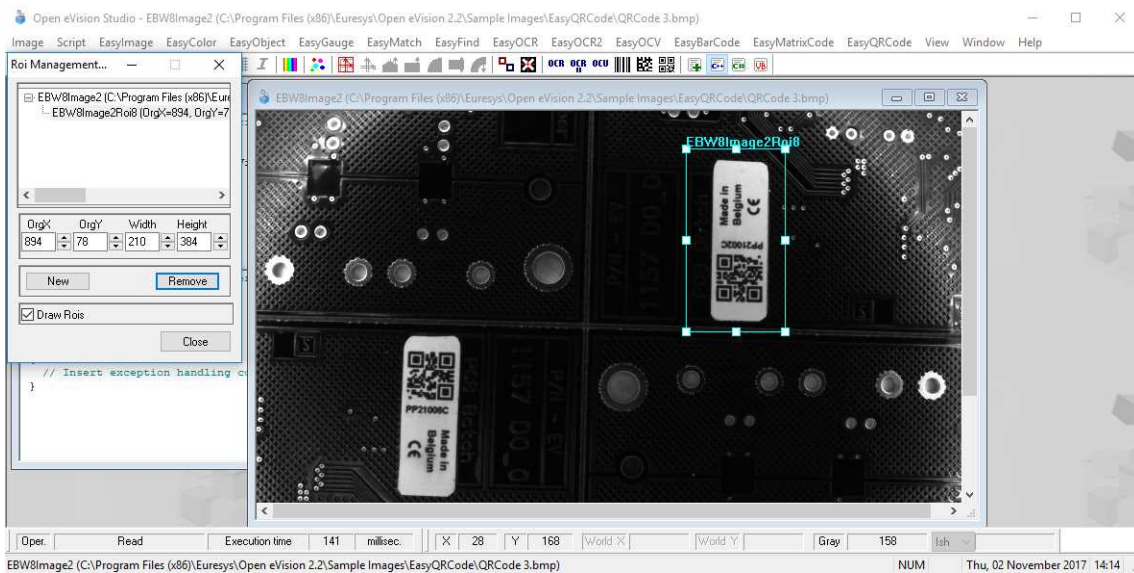
- 이미지가 이미 열려 있으면 해당 이미지 창을 활성화하십시오.
- 이미지가 아직 열리지 않은 경우 기본 메뉴로 이동하십시오: 열기 위해서 **Image > Open...**

- #### 2. ROI를 만들려면 기본 메뉴로 이동하십시오: **Image > ROI Management....**

ROI Management 윈도우는 아래 보이는 대로 표시됩니다.



3. 트리에서 이미지를 선택하십시오.
 4. **New** 버튼을 클릭하십시오.
 5. 대화 상자에서 새 RIO를 위한 **Variable name** 를 넣으십시오.
- ROI는 아래 그림과 같이 이미지에 색상 사각형으로 표시됩니다.



6. ROI 모서리와 측면 핸들을 끌어 필요한 위치로 이동하십시오.
 7. ROI Management 윈도우를 닫기 위해 **Close** 버튼을 클릭하십시오.
- 다음 단계는 "4 단계 : 도구 구성하기" 다음 페이지.

ROI 관리하기

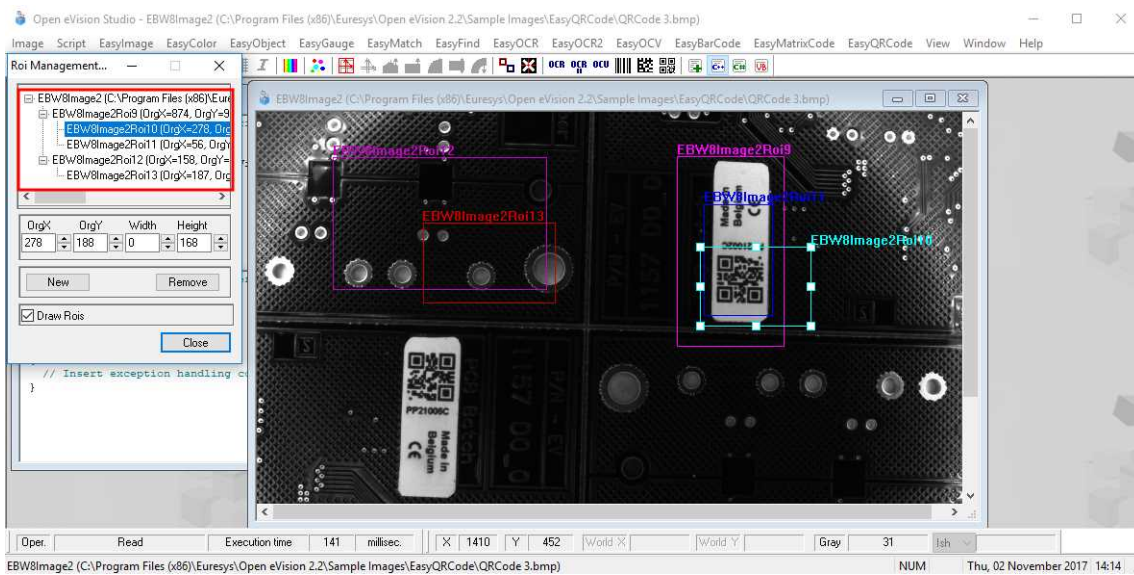
ROI를 추가, 변경 및 제거할 수 있습니다.

이미지는 여러 개의 ROI를 가질 수 있습니다. 각 ROI는 이미지에 직접 첨부 될 수 있습니다(즉, 위치가 이미지와 관련됨) 또는 다른 ROI (이 위치가 '부모' ROI와 관련이 있음을 의미함)에 첨부할 수 있습니다.

1. ROI를 관리하려면, 기본 메뉴로 이동하십시오: **Image > ROI Management...**

ROI Management 윈도우가 아래와 같이 ROI 관계 트리와 함께 표시됩니다.

Draw Rois 상자가 선택되면 모든 ROI가 이미지에 다른 색상으로 표시됩니다.



2. ROI 관계 트리에서 ROI를 선택하십시오.
3. ROI 모서리와 측면 핸들을 드래그하여 선택한 ROI의 위치와 크기 (해당하는 경우 모든 ROI의 위치)를 변경하십시오.
4. **New** 클릭하면 선택한 ROI에 첨부 된 새로운 ROI를 추가 할 수 있습니다.
ROI를 이미지에 직접 첨부하려면 ROI 관계 트리의 상단에서 이미지를 선택하십시오.
5. 선택한 ROI (있는 경우 모든 해당 ROI가 첨부됩니다)를 삭제하려면 **Remove** 버튼을 클릭하십시오.
6. ROI Management 윈도우를 닫기위해 **Close** 버튼을 클릭하십시오.

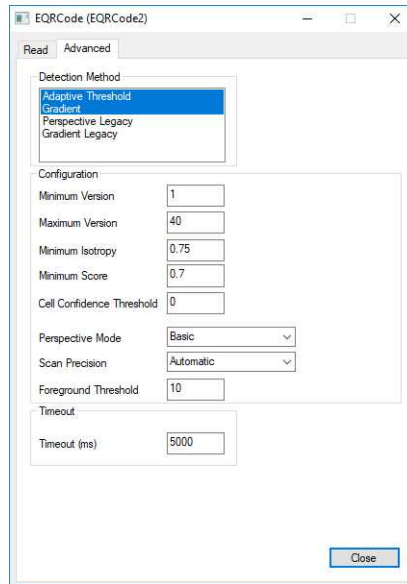
4 단계 : 도구 구성하기

일단 ROI를 작성했다면 ROI를 포함하여 이미지가 준비되면 도구를 구성해야 합니다.

도구 윈도우에서:

1. 여러 탭을 여십시오.

새 도구를 생성하면 모든 매개 변수가 기본값으로 설정됩니다.



EasyQRCode 도구의 매개 변수 탭 예제

2. 각 탭에서 원하는대로 매개 변수의 값을 설정하십시오.

매개 변수, 기능 및 기본값에 대한 자세한 내용은 "기능 안내서" 및 "레퍼런스 매뉴얼"을 참조하십시오.

학습 또는 게이지 사용과 같은 특정 작업에 대해서는 "기능 안내서"를 참조하십시오.

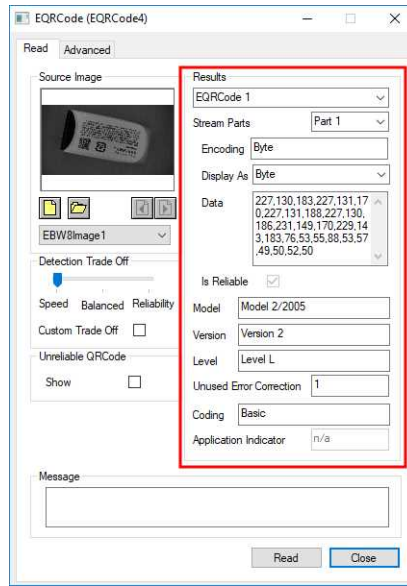
3. 다음 단계에서 설명한대로 도구를 실행하고 결과를 분석하십시오 **"5 단계 : 도구 실행 및 실행 시간 확인하기"** 아래.

5 단계 : 도구 실행 및 실행 시간 확인하기

도구 매개 변수가 설정되면 도구를 실행하고 원하는 경우 컴퓨터에서 실행 시간을 확인하십시오.

도구 윈도우에서:

1. 선택한 이미지에서 도구를 실행하려면 **Read**, **Detect**, **Results** 혹은 **Execute** 버튼을 클릭하십시오.(라이브러리 기능에 따라 다름)
2. 아래 그림과 같이 이미지와 결과 필드 또는 영역에서 결과를 확인하십시오.

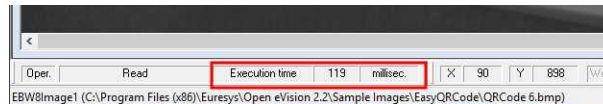


QRCode를 읽은 결과의 예

3. 예상되는 결과가 없는 경우:

- 매개 변수를 변경하십시오 (기본값으로 시작한 다음 한 번에 하나의 매개 변수만 변경하십시오).
- 이미지가 충분하지 않은 경우, .설명 된대로 이미지를 강화시키십시오.

4. 기본 Open eVision Studio윈도우의 왼쪽 하단에 있는 실행 시간 막대에서 실행 시간을 확인하십시오.



실행 시간

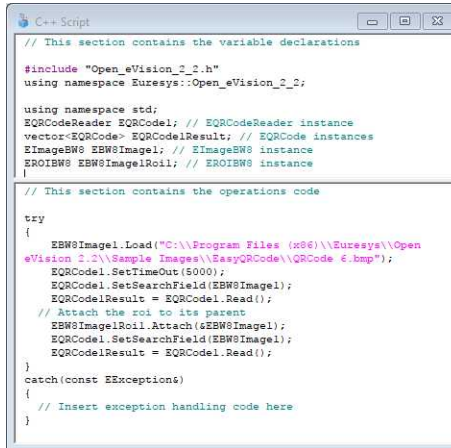
실행 시간은 컴퓨터에서 측정한 실제 처리 시간입니다. 그것은 컴퓨터 프로세서, 메모리, 운영 체제 및 실행시 프로세서 로드 에 따라 다릅니다. 따라서 이 실행 시간은 실행마다 약간씩 다릅니다.

- 5. 보다 대표적인 실행 시간을 얻으려면 **Read**, **Detect**, **Results** 혹은 **Execute** 여러 번 클릭하고 평균 실행 시간을 계산하십시오.
- 6. 애플리케이션에서 실행 시간을 줄여야 하는 경우 다음을 시도하십시오.
 - 도구 매개 변수를 변경하려면,
 - 이미지에 하나 이상의 ROI를 추가하려면,
 - 이미지 향상하기 위해

다음 단계는 "6단계: 생성된 코드 사용하기" 다음 페이지.

6단계: 생성된 코드 사용하기

기본적으로 Open eVision Studio 인터페이스에서 수행하는 모든 작업을 아래 그림과 같이 선택한 언어의 코드로 변환합니다.



```

C++ Script
// This section contains the variable declarations
#include "Open_eVision_2_2.h"
using namespace Euresys::Open_eVision_2_2;

using namespace std;
EQRCoderReader EQRCoder; // EQRCoderReader instance
vector<EQRCoder> EQRCoderResult; // EQRCoder instances
EImageBW8 EBW8Image1; // EImageBW8 instance
EROIBW8 EBW8Image1RoI; // EROIBW8 instance
|
// This section contains the operations code
try
{
    EBW8Image1.Load("C:\\Program Files (x86)\\Euresys\\Open
eVision 2.2\\Sample Images\\EasyQRCode\\QRCode_6.bmp");
    EQRCoder.SetTimeout(5000);
    EQRCoder.SetSearchField(EBW8Image1);
    EQRCoderResult = EQRCoder.Read();
    // Attach the roi to its parent
    EBW8Image1RoI.Attach(EBW8Image1);
    EQRCoder.SetSearchField(EBW8Image1);
    EQRCoderResult = EQRCoder.Read();
}
catch(const EExceptions&)
{
    // Insert exception handling code here
}

```

도구 결과가 적합하면 이 생성된 코드를 저장하거나 복사하여 자신의 애플리케이션에서 사용할 수 있습니다.

애플리케이션에 코드 복사 및 붙여 넣기

스크립트 윈도우에서:

1. 복사할 코드 섹션을 선택하십시오.
2. 이 코드에서 오른쪽 클릭하고 메뉴에서 **복사** 를 클릭하십시오.
3. 개발 환경 도구로 가서 코드를 제자리에 붙여 넣으십시오.

코드를 저장하십시오.

1. 스크립트 메뉴로 가십시오.
2. 다른 스크립트로 저장...을 클릭하십시오.
3. 코드를 텍스트 파일로 저장하려면 파일 이름과 경로를 입력하십시오.

생성된 코드 관리하기

스크립트 메뉴에서:

- 프로그래밍 언어를 선택하십시오 (언어를 변경하면 스크립트 창 콘텐츠가 자동으로 삭제됩니다).
- 스크립트 코드 생성을 활성화하거나 비활성화하십시오. 코드로 저장하지 않고 일부 작업을 수행하려면 이 옵션을 비활성화하십시오.

1.4. 이미지 전처리 및 저장

언제 이미지를 사전 처리해야 하나요?

물론, 최적의 상황은 이미지를 쉽고 잘 처리 할 수 있도록 이미지 수집 시스템을 설정하여 Open eVision 도구를 원활하고 효율적으로 실행하는 것입니다.

이것이 가능하지 않거나 쉽게 할 수 없는 경우 실행하고 싶은 Open eVision 도구를 위해서 이미지 또는 ROI를 사전 처리하여 강조하고 대비해 둘 수 있습니다.

다양한 사용 가능한 기능을 사용하여 이미지의 이득과 오프셋을 조정하고, 컨볼루션을 적용하고, 임계값을 조정하고, 이미지를 회전 및 화이트 밸런스 조정하고, 윤곽선을 강조할 수 있습니다... EasyImage 와 EasyColor 기능을 사용합니다.

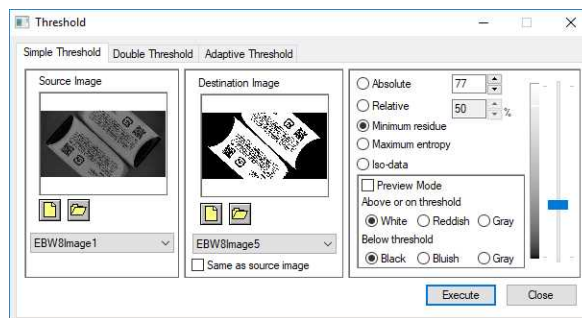
이미지 전처리

이미지 전처리와 실행중인 도구의 차이점은 전처리가 새로운 이미지를 생성하는 반면 도구는 주로 이미지를 변경하지 않고 이미지에서 추출 및 검색한다는 점입니다.

이미지나 ROI 전처리하기:

1. 기본 메뉴 표시 줄에서 사용하려는 라이브러리 (EasyImage 또는 EasyColor)를 클릭하십시오.
2. 사용할 기능을 클릭하십시오.

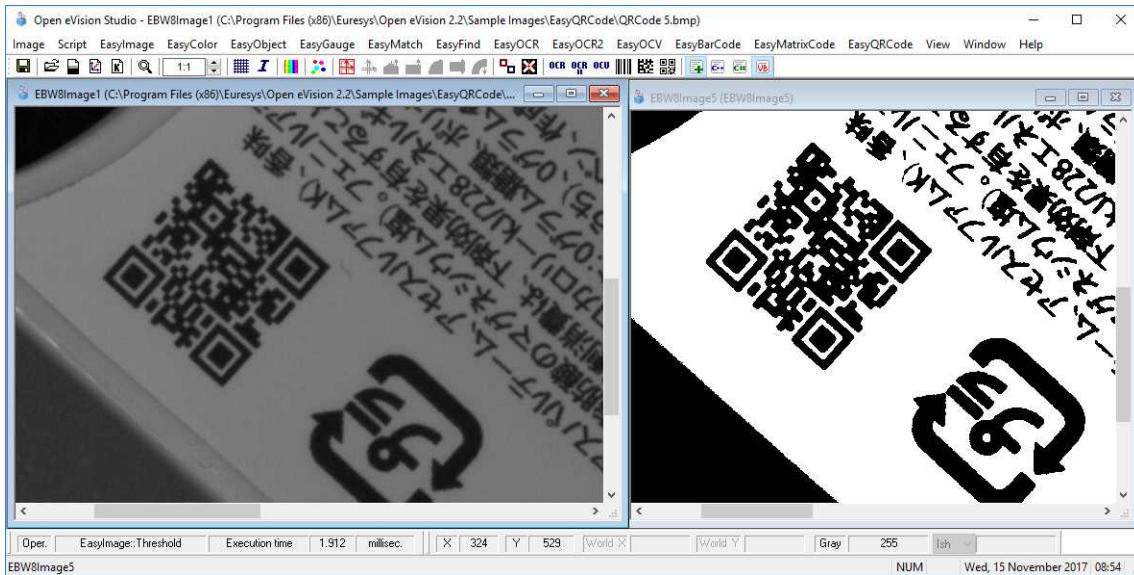
대부분의 기능 대화 상자는 아래 그림과 같이 2 개의 이미지 선택 영역과 매개 변수 설정 영역으로 비슷하게 되어 있습니다.



사전 처리 대화 상자의 예 (EasyImage와 임계값)

3. 선택한 기능에 여러 버전이 있는 경우 해당 탭을 엽니다.
4. 소스 이미지 영역에서, 소스 이미지를 엽니다. ("2단계 : 이미지 열기" 페이지10에서 설명한 대로).
5. 대상 이미지 영역에서, 새로운 대상 이미지를 열거나 만드십시오.
6. 매개 변수를 설정하십시오.
7. 실행 버튼을 클릭하십시오.

아래 그림과 같이 대상 이미지에서 전처리된 이미지를 볼 수 있습니다.



소스 이미지와 대상 이미지 (EasyImage와 임계값)

8. Open eVision Studio외의 다른 곳에서 대상 이미지를 사용하고 싶으면, 아래 설명한 대로 이미지를 저장하십시오.

이미지 저장하기

1. 저장하려는 이미지를 클릭하여 활성화하십시오.
2. 저장 메뉴를 열려면 다음 중 하나를 수행하십시오.
 - 이미지에서 마우스 오른쪽 버튼을 클릭하십시오.
 - 혹은 기본 메뉴> 이미지를 여십시오.
3. 다른 이름으로 저장...을 클릭하십시오.
4. 파일 형식을 선택하십시오.(JPEG, JPEG2000, PNG, TIFF 또는 비트맵)
5. 파일 이름을 적고 경로를 선택하십시오.
6. 저장 버튼을 클릭하십시오.

2. Tutorials

2.1. Preliminaries

About the application basic interface, operations and workflow, please refer to "[Getting Started](#)" on page 1, available through the **Help** menu. About the Open eVision functional concepts, and about the API classes, methods, properties, refer to the main section "[Using Open eVision](#)" on page 1.

There are often many different ways to get the same results through the graphical interface. The tutorials show only one way for each case.

All the images used in the tutorials are located by default in the **\Development Tools\Images** subdirectory available in the Open eVision installation directory. As this root path is constant, it is not repeated in each tutorial.

By default, in the main toolbar, the **Activate/Deactivate script code generation** toggle button is on. This will insert automatically into the script window the code corresponding to the actions that are taken through the graphical interface. But note that the automatic operations are not inserted into the script until they have been explicitly performed by clicking a dialog box button (**Execute**, **Find**, **Learn...**).

2.2. EasyImage

Converting a Gray-Level Image into a Binary Image

Thresholding

[Single Thresholding](#) - [Double Thresholding](#) - [Histogram-Based Single Thresholding](#) - [Histogram-Based Double Thresholding](#)

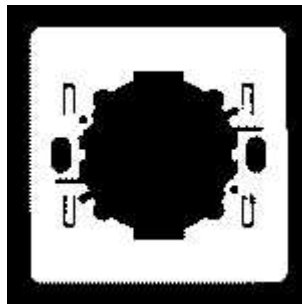
Objective

Following this tutorial, you will learn how to use EasyImage to convert a gray-level source image into a binary destination image. Thresholding an image transforms all the gray pixels into black or white pixels, depending on whether they are below or above a specified threshold. Thresholding an image makes further analysis easier.

You'll need first to load an image (step 1). Then you'll set the thresholding parameters (step 2), and perform the conversion (step 3).



Gray-level source image



Black and white destination image, after thresholding

Step 1: Load the source image

1. From the main menu, click **EasyImage**, then **Threshold**.
2. In the **Simple Threshold** tab, click the **Open** icon of the Source Image area, and load the image file `EasyMatch\Switch1.tif`.
3. Keep the default variable name for the new Image object, and click **OK**.

Step 2: Set the thresholding parameters

1. In the right area of the **Threshold** dialog box, move the slider to change the threshold, and see directly in the source image a preview of the result.
2. Select the **Minimum residue** option to set a pre-defined algorithm that finds automatically the right threshold.

Step 3: Perform the conversion

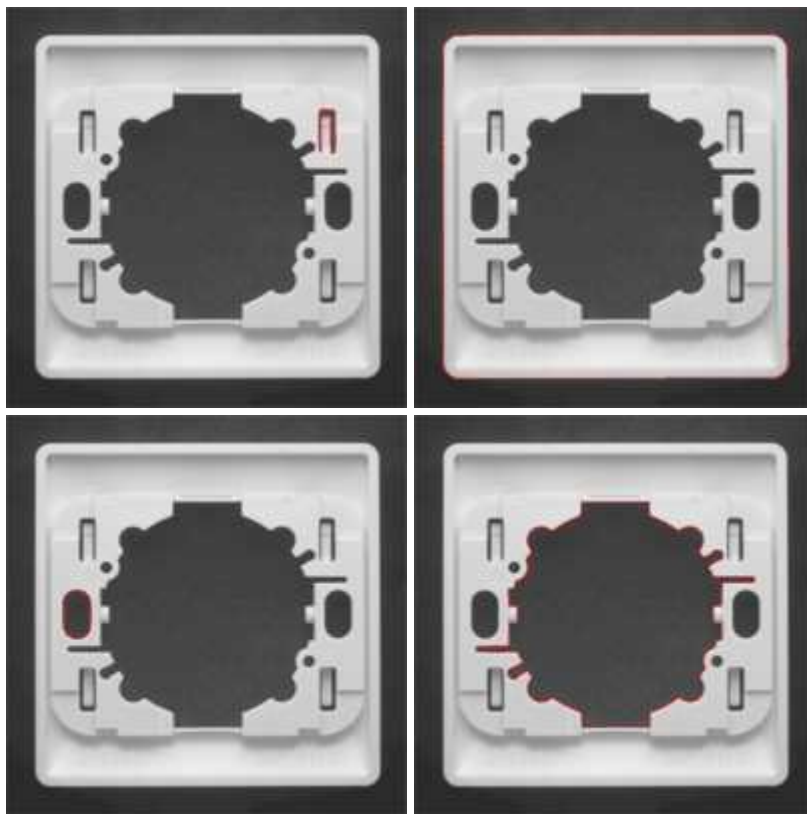
1. Click the **New** icon in the **Destination Image** area to create a new destination image.
2. Keep the default settings for the new Image object, and click **OK**.
3. In the **Threshold** dialog box, click **Execute** to perform the thresholding in the destination image.

Extracting an Object Contour

Objective

Following this tutorial, you will learn how to use EasyImage to trace an object outline in a gray-level image. The contour extraction allows you to get in a path vector all the points that constitute an object contour, just by clicking an edge of this object.

You'll need first to load an image (step 1) and set a vector that will contain all the contour points (step 2). Then you'll click an object edge, and the contour will be extracted automatically (step 3).



Contours are extracted from object edges

Step 1: Load the source image

1. From the main menu, click **EasyImage**, then **Contour**.
2. Click the **Open** icon of the Source Image area, and load the image file `EasyMatch\Switch1.tif`.
3. Keep the default variable name for the new image object, and click **OK**.

Step 2: Set the destination vector

1. Click the **New** icon in the Destination Vector area.
2. Keep the default settings and variable name for the new vector object.
3. Click **OK**.

Step 3: Extract the contour

1. When moving the cursor above the image, an arrow appears.
2. Move the arrow above an object edge, and click.

From this object edge, a contour is traced, and a red line appears around the object.

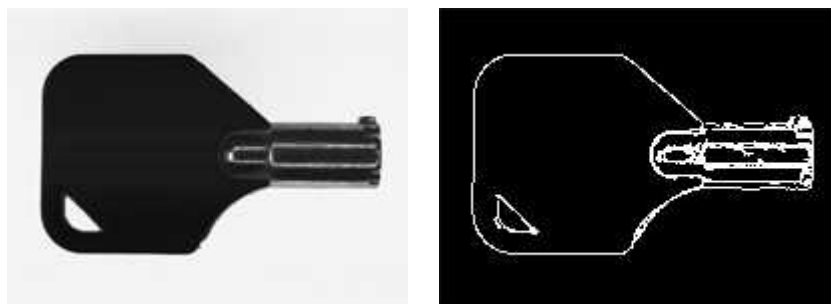
The destination vector is filled with the points constituting the contour.

Transforming a Gray-Level image into its Black and White Edges

Objective

Following this tutorial, you will learn how to use EasyImage to transform a gray-level image to a binary image, keeping only the edges detected in the image. The conversion uses the Canny edge detector algorithm.

You'll need to load a source image (step 1), and simply apply the Canny edge transformation (step 2).



Source image (left) and destination image after Canny edge transformation (right)

Step 1: Load the source image

1. From the main menu, click **EasyImage**, then **Canny Edge Detector**.
2. Keep the default variable name, and click **OK**.
3. Click the **Open** icon of the Source Image area, and load the image file `EasyImage\Key1.tif`.

4. Keep the default variable name, and click **OK**.

Step 2: Apply the Canny edge transformation

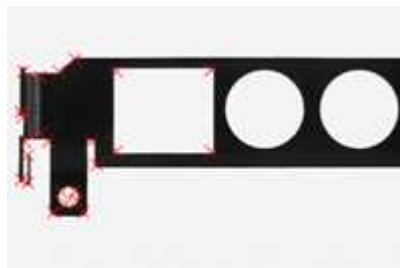
1. Click the **New** icon in the Destination Image area to create a new destination image.
2. Keep the default settings, and click **OK**.
3. In the canny edge detector dialog box, click **Apply** to perform the operation in the destination image.

Detecting the Corners of an Object Using Harris Corner Detector

Objective

Following this tutorial, you will learn how to use EasyImage to detect the corners of an object. The detection uses the Harris corner detector algorithm.

You'll need to load a source image (step 1), and simply apply the Harris corner detection (step 2).



Corners are detected in the source image

Step 1: Load the source image

1. From the main menu, click **EasyImage**, then **Harris Corner Detector**.
2. Keep the default variable name, and click **OK**.
3. Click the **Open** icon of the Source Image area, and load the image file `EasyGauge\Bracket1.tif`.
4. Keep the default variable name, and click **OK**.

Step 2: Apply the Harris corner detection

1. In the Harris corner detector dialog box, enter 2.3 for the Scale property.
2. Click **Apply** to perform corners detection.

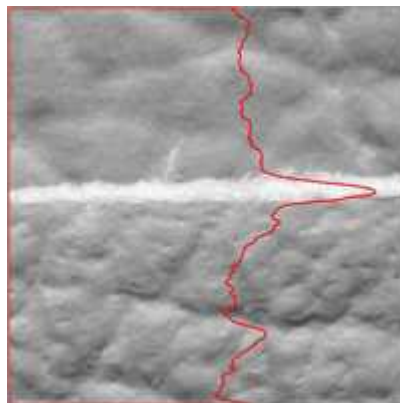
3. Click **Results** to display the coordinates of all detected corners.
4. The **Columns** button allows you to display additional properties in the results list.

Detecting a Horizontal or Vertical Line Using Projection

Objective

Following this tutorial, you will learn how to use EasyImage to detect defects (horizontal/vertical line) in a gray-level image.

You'll need first to load a source image (step 1), set a vector (step 2), and then detect the defect (horizontal line) (step 3).



Defects can be detected using the image projection

Step 1: Load the source image

1. From the main menu, click **EasyImage**, then **Projection**.
2. Click the **Open** icon of the Source Image area, and load the image file `EasyImage\Leather.bmp`.
3. Keep the default variable name for the new image object, and click **OK**.

Step 2: Set the destination vector

1. Click the **New** icon in the Destination Vector area.
2. Select the BW32 option for the vector type, and click **OK**.

Step 3: Detect the defects

1. In the Image Projection dialog box, select the **column** button, and click **Execute** to perform the operation.
2. The resulting vector and the corresponding plot are displayed in the destination vector window. The graphical result also appears on the image. Each vector value is the sum of all pixels values across the corresponding horizontal row (or vertical column). By this mean, horizontal (or vertical) defects are easily detected.

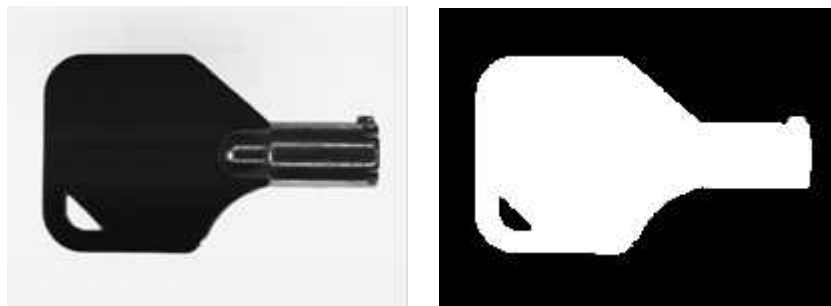
Creating a Flexible Mask

Using Flexible Masks

Objective

Following this tutorial, you will learn how to create a flexible mask from a source image, to restrict a future processing to an arbitrary-shaped do-care area.

Flexible masks can be created in any ways to build a bi-level image. Here, we will first load the source image (step 1), and then successively invert it, and threshold it (steps 2-3). The resulting image—the flexible mask— will be saved as a new image file (step 4). This new image file is a bi-level image. However, there are still black areas that need to be erased, before using the image as a flexible mask. You can use a third-party software, such as Paint, to clear the unwanted areas.



Source image (left) and flexible mask image (right)

Step 1: Load the source image

1. From the main menu, click **EasyImage**, then **Arithmetic & logic**.
2. Click the **Open** icon of the Source Image 0 area, and load the image file `EasyImage\Key1.tif`.
3. Keep the default variable name for the new image object, and click **OK**.

Step 2: Invert the image

1. Click the **New** icon of the **Destination** area.
2. Keep the default settings for the new image object, and click **OK**.

3. In the **Operation** drop-down list, select **Invert**, and click **Execute**

Step 3: Threshold the image

1. From the main menu, click **EasyImage**, then **Threshold**.
2. In the Source Image area, select the inverted image from the drop-down list.
3. Click the **New** icon of the Destination area.
4. Keep the default settings for the new image object, and click **OK**.
5. Select the Absolute option, enter '46' as the threshold value, and click **Execute**.

Step 4: Save the flexible mask

1. Right-click in the destination image, and select **Save As...**
2. Type a file name for the new flexible mask file. Finally, click **Save**.

The new image now is a bi-level image. However, there are still black areas that need to be erased, before using the image as a flexible mask. You can use a third-party software, such as Paint, to clear the unwanted areas.

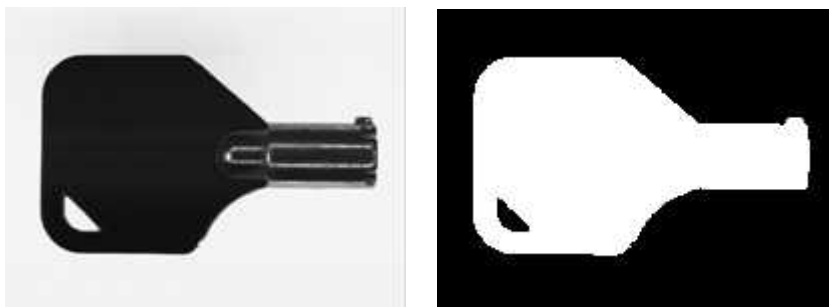
Computing Gray-Level Statistics Using a Flexible Mask

Using Flexible Masks

Objective

Following this tutorial, you will learn how to compute gray-level statistics on an arbitrary-shaped area only.

You'll need first to load a source image (step 1), and a flexible mask image (step 2). The mask image must be applied on the source image (step 3), to separate do-care areas (that must be considered) and don't-care areas (that should not be considered). Finally, the gray-level statistics are computed on the do-care area only (step 4).



Source image (left) and flexible mask image (right)

Step 1: Load the source image

1. From the main menu, click **EasyImage**, then **Image Statistics, Gray Scale**.
2. Click the **Open** icon of the Source Image area, and load the image file *EasyImage\Key1.tif*.
3. Keep the default variable name for the new image object, and click **OK**.

Step 2: Load the flexible mask image

1. From the main menu, click **Image**, then **Open....**
2. Load the image file *EasyImage\Mask2.bmp*.
3. Keep the default variable name for the new image, and click **OK**.

Step 3: Apply the flexible mask on the source image

1. In the Mask area of the **Gray Scale Image Pixels Statistics** dialog box, select the mask image from the drop-down list.
2. The source image preview in the dialog box shows (in red diagonal lines) the don't-care area, that is the area that will be not be considered when computing the gray-level statistics.

Step 4: Compute the gray-level statistics

1. Select the **Pixel Count** check-box.
2. Click **Execute**.

The results are displayed in the read-only fields below.

Detecting the Corners of an Object Using Hit-and-Miss Transform

Hit-and-Miss Transform

Objective

Following this tutorial, you will learn how to use EasyImage to detect top corners in an image, using the hit-and-miss transform.

You'll need to load a source image (step 1), set the kernel that represents a top corner (step 2), and then set a destination image and simply execute the hit-and-miss transform (step 3).



Source image (left) and top corner detected in the source image (white dot)

Step 1: Load the source image

1. From the main menu, click **EasyImage**, then **Hit And Miss**.
2. Click the **Open** icon of the Source Image area, and load the image file EasyImage\Diamond.bmp.
3. Keep the default variable name, and click **OK**.

Step 2: Set the hit-and-miss kernel

- In the **Hit And Miss** dialog box, set the kernel according to the following values:

	-1	0	1
-1	Don'tCare	Background	Don'tCare
0	Background	Foreground	Background
1	Foreground	Foreground	Foreground

Kernel that detects top corners

Step 3: Apply the hit-and-miss transform

1. Click the **New** icon of the **Destination Image** area.
 2. Keep the default parameters and variable name, and click **OK**.
 3. Click **Execute** to perform the operation.
- The top corner (white dot) is detected.
4. Try with other kernel configurations to detect the other corners.

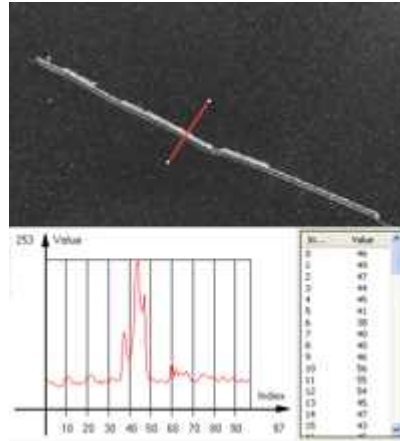
Extracting a Vector Using Profile Function

Profile Sampling

Objective

Following this tutorial, you will learn how to use EasyImage to detect scratches.

You'll need first to load an image (step 1), set a destination vector, and detect the scratches (step 2).



Scratches can be detected using a profile

Step 1: Load the source image

From the main menu, click **EasyImage**, then **Profile**.

1. Click the **Open** icon of the **Source Image** area, and load the image file `EasyImage\Plastic.tif`.
2. Keep the default variable name for the new image object, and click **OK**.

Step 2: Set the destination vector and detecting the scratches

1. Click the **New** icon in the Destination Vector area.
2. Select the **BW8** option for the vector type, and click **OK**.
3. A profile appears on the image (red line segment). In the destination vector window, vector values correspond to pixels along the line segment.

The scratch is detected as a sharp deviation in the vector graph.

4. Using the mouse, drag the handles to move or resize the red line segment, and observe the plot evolution.

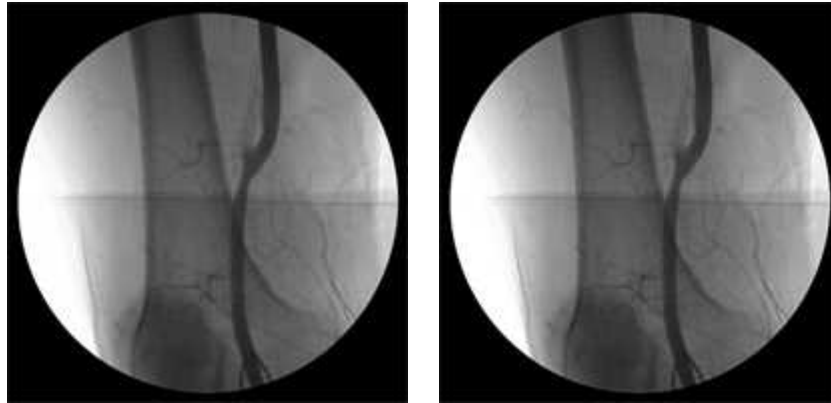
The sharp deviation appears whenever the line segment is placed across the scratch.

Enhancing an X-ray image

Objective

Following this tutorial, you will learn how to use EasyImage to enhance an X-ray image.

You'll need first to load an image (step 1), then define convolution parameters to enhance the image (step 2).



Source image (left) and enhanced image, after predefined and user-defined convolutions (right)

Step 1: Load the source image

1. From the main menu, click **EasyImage**, then **Convolution**.
2. Click the **Open** icon of the **Source Image** area, and load the image file `EasyImage\XRay.bmp`.
3. Keep the default variable name for the new image object, and click **OK**.
4. Click the **New** icon of the **Destination Image** area to create a new destination image.
5. Keep the default variable name and click **OK**.

Step 2: Set the convolution parameters

1. From the **Predefined kernels** drop-down list, select **Highpass2**, and click **Execute** to perform the operation.

The image is no longer blurred but the result is bad because the filter has revealed the noise of the source image. We need to create a new convolution kernel that will apply a softer high-pass filtering.

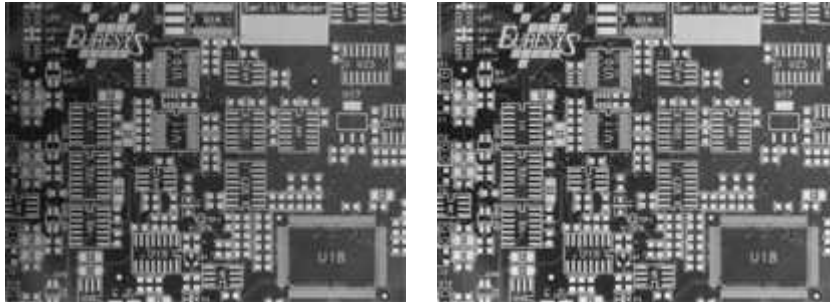
2. Click the **New** icon next to the **User defined kernels** drop-down list.
3. Keep the default dimension (3x3) and variable name, and click **OK**.
4. Enter the following kernel data from left to right and top to bottom: -1, -1, -1; -1, 15, -1; -1, -1, -1, and click **Apply**.
5. Click **Execute** in the Convolution dialog box to perform the operation. The image is much clearer now.

Correcting Non-Uniform Illumination

Objective

Following this tutorial, you will learn how to use EasyImage to correct non-uniform illumination in an image.

You'll need first to load an image (step 1), load a light reference image (step 2), and perform the correction (step 3).



Source image, with non-uniform illumination (left) and corrected image (right)

Step 1: Load the source image

1. From the main menu, click **EasyImage**, then **Uniformize**.
2. Click the **Open** icon of the Source Image area, and load the image file `EasyImage\Board (original).tif`.
3. Keep the default variable name for the new image object, and click **OK**.

Step 2: Load the reference image

1. Click the **Open** icon of the **Light Reference** area, and load the image file `EasyImage\Board (light reference).tif`.

To obtain the light reference image, we used a white screen illuminated in the same condition as the board (original image).

2. Keep the default variable name for the new image object, and click **OK**.

Step 3: Perform the correction

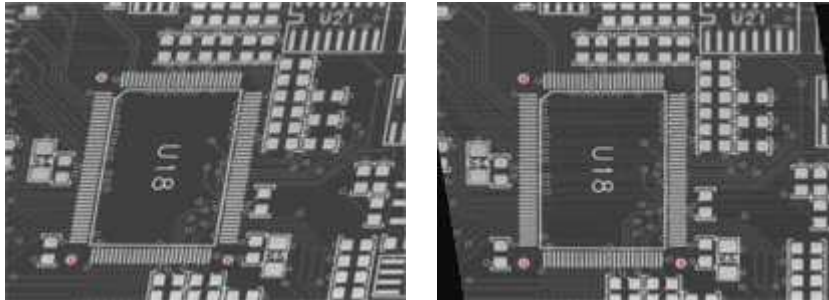
1. Click the **New** icon in the **Destination Image** area to create a new destination image.
2. Keep the default values and click **OK**.
3. Click **Execute** to perform the operation.
4. In both source and destination images, right-click and select **3D Rendering**.
5. In the new 3D windows, click and drag the mouse to rotate the view. Compare the profiles.

Correcting Shear Effect

Objective

Following this tutorial, you will learn how to use EasyImage to correct a shear effect in an image. The following image is taken by a line-scan camera. The camera sensor was misaligned, resulting in a so-called shear effect.

You'll need first to load an image (step 1), create a destination image (step 2), and then set pivots parameters to perform the correction (step 3).



Source image, with a shear effect (left) and corrected image (right)

Step 1: Load the source image

1. From the main menu, click **EasyImage**, then **Register**.
2. Click the **Open** icon of the Source Image area, and load the image file `EasyImage\Shear.tif`.
3. Keep the default variable name for the new image object, and click **OK**. Three pivots points appear in the image.

Step 2: Create a destination image

1. Click the **New** icon of the **Destination Image** area.
2. Enter '768' and '576' as image width and height, and click **OK** to accept the default name. Three pivots points appear in the image.

Step 3: Set the pivots parameters

1. In the source image, using the mouse, drag each pivot to the center of the fiducial marks (the dots around the U18 area).

Notice that the source pivots coordinates, in the Register dialog box, have changed accordingly.

2. To correct the image, enter the following destination pivots coordinates:
 - X0: 170
 - Y0: 495
 - X1: 470
 - Y1: 495
 - X2: 170

□ Y2: 144

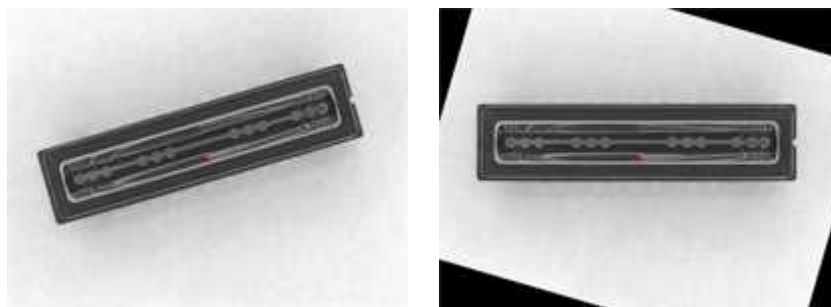
3. Click **Execute** to perform the operation.

Correcting Skew Effect

Objective

Following this tutorial, you will learn how to use EasyImage to correct skew effect in an image.

You'll need first to load an image (step 1), create a destination image (step 2), and then set a correction angle to perform the correction (step 3).



Source image, with a skew effect (left) and corrected image (right)

Step 1: Loading the source image

1. From the main menu, click **EasyImage**, then **Scale and Rotate**.
2. Click the **Open** icon of the **Source Image** area, and load the image file `EasyImage\CCD.tif`.
3. Keep the default variable name for the new image object, and click **OK**.

Step 2: Creating a destination image

1. Click the **New** icon of the **Destination Image** area.
2. Enter '768' and '576' as image width and height, and click **OK** to accept the default name.

Step 3: Setting the correction angle

1. Select the **Rotate** option, and enter -16.17 in the Angle (Deg) field. (To measure this rotation angle, refer to Measuring the rotation angle of an object.)
2. From the Interpolation bits drop-down list, select 8 bits to get a better result.
3. Click **Execute** to perform the operation.

2.3. EasyColor

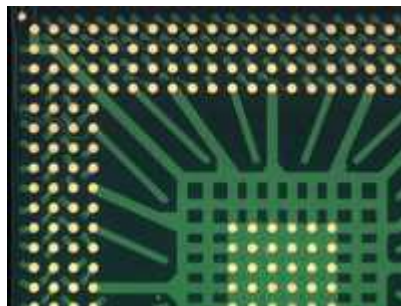
Performing Thresholding on Color Images

Color Components

Objective

Following this tutorial, you will learn how to use EasyColor to segment a color source image, by setting a threshold value for each color component of the current color system. For example, to retrieve the solder pads on a PCB, you'll perform a color segmentation based on the golden pixels (H), with a loose discrimination on the brightness (L) and saturation (S), to eliminate surface and lighting effects.

You'll need first to load a color source image, create a destination image, and a color lookup table (steps 1-3). Then, you'll set the color system and tune each component tolerance to get a satisfying segmentation of the solder pads (step 4).



Source image



Thresholded image

Step 1: Load the source image

1. From the main menu, click **EasyColor**, then **Threshold**.

2. Click the **Open** icon of the **Source Image** area, and load the image file EasyColor\BGA Substrate Color.jpg.
3. Keep the default variable name for the new Image object, and click **OK**.
4. Disable the **Preview Mode** check-box to see the raw source image.

Step 2: Create a destination image

1. Click the **New** icon of the **Destination Image** area.
2. Keep the default variable name for the new Image object, and click **OK**.

Step 3: Create a color lookup table

1. Click the **New** icon of the **Color Lookup** area.
2. Keep the default variable name for the new color lookup object, and click **OK**.

Step 4: Perform the color segmentation

1. Select **LSH** from the **Color System** drop-down list.
2. In the source image, click in a golden pad. The pixel lightness, saturation and hue values are updated in the **Color Threshold** dialog box.
3. Adjust the tolerance of lightness and saturation to enlarge the range of thresholded pixels, until you get a satisfying segmentation in the destination image.

Performing Color Segmentation

Color Components

Objective

Following this tutorial, you will learn how to use EasyImage to perform color segmentation.

You'll need first to load an image (step 1), create a color look-up table (step 2), and perform the segmentation (step 3).



Source image



Segmented image

Step 1: Load the source image

1. From the main menu, click **EasyColor**, then **Threshold**.
2. Click the **Open** icon of the Source Image area, and load the image file `EasyColor\Pills.tif`.
3. Keep the default variable name for the new image object, and click **OK**.
4. Disable the **Preview Mode** check-box to see the raw source image.

Step 2: Create a color lookup table

1. Click the **New** icon of the Color Lookup area.
2. Keep the default variable name for the new color lookup object, and click **OK**.

Step 3: Perform the color segmentation

1. Select **LSH** from the **Color System** drop-down list.
2. In the source image, click in the center of a green pill. The pixel lightness, saturation and hue values are updated in the **Color Threshold** dialog box.
3. Increase the lightness tolerance up to 120. Increase the saturation tolerance up to 50.
4. Enable the **Preview Mode** check-box to see the result of the segmentation. If needed, click in the green pills to improve the result.
5. Click the **New** icon of the Destination Image area.
6. Keep the default settings for the new Image object, and click **OK**.
7. The new image is automatically thresholded. Clicking **Execute** will insert the corresponding code into the script windows.

2.4. EasyObject

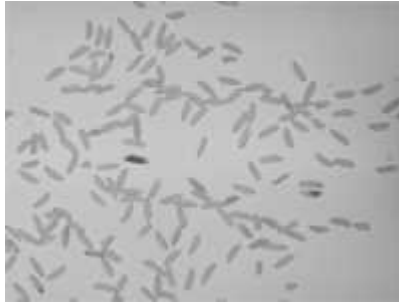
Removing Non-Significant Objects After Image Segmentation

Image Segmenter

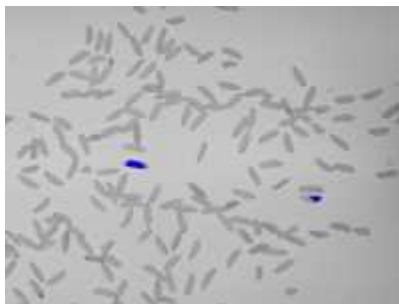
Objective

Following this tutorial, you will learn how to use EasyObject to detect bad rice grains (largely dark) among many normal rice grains (largely light).

You'll need first to load the source image (step 1). Then you'll perform the image segmentation, based on a threshold value (step 2). All the detected objects are dark, but some are too small to be significant. So, you'll set a minimum object area (number of pixels), and remove the smallest objects (step 3). Finally, you'll get only the objects that really represent bad rice grains.



Source image



Bad rice grains are detected

Step 1: Load the source image

1. From the main menu, click **EasyObject**, then **New EasyObject Tool**.
2. Keep the default variable name for the new object, and click **OK**.
3. In the **Encoder** tab, click the **Open** icon of the Source Image area, and load the image file `EasyObject\Rice.jpg`.
4. Keep the default variable name for the new image, and click **OK**.

Step 2: Perform image segmentation

1. In the **Encoder** tab, select the **Black Layer** check-box, and unselect the **White Layer** check-box.
2. Click the ... button around the **Threshold** field. In the Threshold dialog box, select **Absolute**, enter '115', and click **OK**.
3. Click **Encode** to detect the dark objects. In the image, each object is drawn using a different color.

4. Click **Results** to display the list of all the detected objects. Clicking an object in the image highlights it in the list, and vice versa.

Step 3: Remove the smallest objects

1. In the objects list, click **Columns**.
2. Tick the **Area** check-box, and click **OK**. In the list view, a new Area column appears, displaying each object area.
3. Click the Area column header to sort the objects. There are many small objects (area < 100) that may be considered as noise.
4. In the **Selection** tab, select **Area** from the Feature drop-down list. Select '**Less**' from the Mode drop-down list. In the Threshold field, enter '100'.
5. Click **Remove**. All the objects with an area smaller than 100 pixels have been removed from the list. The two remaining objects are the bad rice grains.

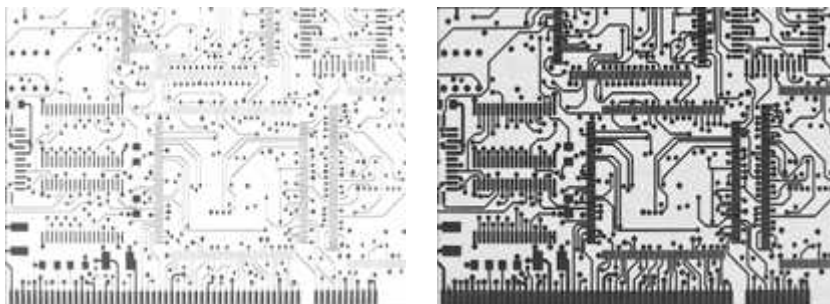
Detecting Differences Between Images Using Min-Max References

Selecting and Sorting Blobs

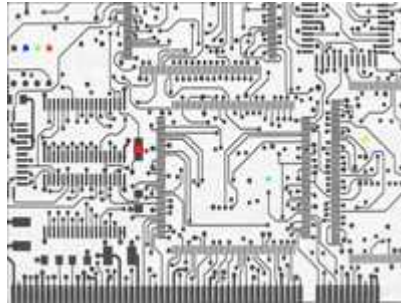
Objective

Following this tutorial, you will learn how to use EasyObject to compare images. In this example, we will check the quality of a PCB film.

You'll need first to load a reliable source image (step 1), from which two reference images (min and max) will be built (step 2). Then you'll load another image to be inspected (step 3), and perform the comparison with the min and max reference images (step 4). The differences will be detected.



High (left) and low (right) threshold reference images



In another image, differences are detected

Step 1: Load the source image

1. From the main menu, click **EasyObject**, then **Make Min Max**.
2. Click the **Open** icon of the Source Image area, and load the image file `EasyObject\FilmOk.png`.
3. Enter 'filmOk' for the name of the new image, and click **OK**.

Step 2: Build min and max reference images

1. Click **Execute**. Min and Max reference images are created, based on the source image.
 - filmOkMax* is computed by dilating *filmOk* a given number of times ('geometric margin') and adding a constant ('gray level margin') to every pixel. *filmOkMin* is computed by eroding *filmOk* the same number of times and subtracting the same constant to every pixel.
 - The geometric margin can be seen as a position tolerance between the image to be inspected and the reference.
 - The gray level margin introduces a tolerance to lighting variations.

Step 3: Load an image to be inspected

1. From the main menu, click **EasyObject**, then **New EasyObject Tool**.
2. Keep the default variable name for the new object, and click **OK**.
3. In the **Encoder** tab, click the **Open** icon of the Source Image area, and load the image file `EasyObject\FilmBad.png`.
4. Enter '*filmBad*' for the name of the new image, and click **OK**

Step 4: Compare the image with the reference images

1. In the **Encoder** tab, select **ImageRange** in the segmentation method drop-down list.
2. Disable the **White Layer** check-box, and enable the **Black Layer** check-box.
3. Click the ... button around the High Image field. Select **filmOkMax** in the drop-down list, and click **OK**.

4. Click the ... button around the Low Image field. Select **filmOkMin** in the drop-down list, and click **OK**.
5. Click **Encode**. Eight differences are highlighted in the image.
6. Click **Results** to get further information about the detected objects. They may be further filtered and analyzed using object features selection and sorting capabilities of EasyObject.

Detecting Printing Errors Using a Flexible Mask

Generating a Flexible Mask from an Encoded Image

Objective

Following this tutorial, you will learn how to use a flexible mask to target and search specific areas in the image.

You'll need first to load a source image (step 1), and a flexible mask image (step 2), that can be automatically applied on the source image to separate do-care areas (that must be considered) and don't-care areas (that should not be considered). Then, you'll perform the inspection only on do-care areas (step 3).



Source image



Mask image



Results

Step 1: Load the source image

1. From the main menu, click **EasyObject**, then **New EasyObject Tool**.
2. Keep the default variable name for the new object, and click **OK**.
3. In the **Encoder** tab, click the **Open** icon, and load the image file `EasyObject\Mobile3.jpg`.
4. Keep the default variable name for the new image, and click **OK**.

Step 2: Load the flexible mask image

1. In the **Encoder** tab, click the **Open** icon, and load the flexible mask image file `EasyObject\MobileMask.bmp`.
2. Enter 'mask' for the name of the new image, and click **OK**. In the **Mask** area of the **Encoder** tab, notice that the mask image is selected from the drop-down list: the mask is automatically applied on the source image, because its name contains 'mask', and because it has been loaded from the coded image dialog box. The source image preview in the dialog box shows (in red diagonal lines) the don't-care area, that is the area that will not be considered when encoding the source image.

Step 3: Inspect the image

1. In the **Segmentation** area of the **Encoder** tab, click the ... button to display the threshold dialog box.
2. Select **Absolute** and enter 202 for the threshold. Click **OK** to close the dialog box.
3. Click **Encode** to locate the objects (in the do-care areas only). In the source image, each object is drawn using a different color. Three printing errors can be observed:
 - The digit '7' is partially printed.
 - The '+' sign is missing.
 - The handset is printed on a lighter tone.
4. Click **Results** to display the statistics on each object.
 - Selecting an object in the list highlights it in the image.
 - Selecting **Columns** and **Drawing** will display more options.

2.5. EasyMatch

Learning a Pattern and Creating an EasyMatch Model File

Pattern Learning

Objective

Following this tutorial, you will learn how to use EasyMatch to learn a model from a reference image, and save it as an EasyMatch model file.

You'll need first to load the reference image (step 1). Then, you'll learn it as the reference model (step 2). And you'll save the model as an EasyMatch model file (step 3).



Reference image

Step 1: Load the reference image

1. From the main menu, click **EasyMatch**, then **New Match Tool**.
2. Keep the default variable name for the new matcher object, and click **OK**.
3. In the **Learning** tab, click the **Open** icon, and load the image file `EasyMatch\FrameModel.tif`.
4. Keep the default variable name for the new image object, and click **OK**.

Step 2: Learn the reference image

- In the **Learning** tab, click **Learn** to acquire the model pattern.

Step 3: Save the model file

1. In the **Learning** tab, click the **Save As...** button.
2. Type a file name for the new EasyMatch model file. Its extension will be `.mch`.
3. Click **Save**.

Matching a Pattern According to a Model File

Pattern Matching and Retrieving Results

Objective

Following this tutorial, you will learn how to use EasyMatch to load an EasyMatch model file, and search for occurrences of the pattern in an image.

You'll need first to load the model file, and a source image where the model will be searched for (steps 1-2). Then the pattern matching is fully automatic (step 3).



Occurrences of the model are automatically highlighted

Step 1: Load the model file

1. From the main menu, click **EasyMatch**, then **New Match Tool**.
2. Keep the default variable name for the new matcher object, and click **OK**.
3. In the **Learning** tab, click **Load...** to open the model file `EasyMatch\Switch.MCH`. The model contains all necessary information about the pattern we are searching for.

Step 2: Load a source image

1. In the **Matching** tab, click the **Open** icon, and load the image file `EasyMatch\Switch1.tif`.
2. Keep the default variable name for the new image object, and click **OK**.

Step 3: Perform the pattern matching

1. The pattern matching is automatically performed on the source image, and the matching occurrences are highlighted. Clicking **Execute** will insert the corresponding code into the script windows.
2. Further information about each occurrence can be found by clicking **Results**.
3. Click in a row to see the corresponding occurrence highlighted in the image.

Learning a Pattern According to an ROI

Pattern Learning

Objective

Following this tutorial, you will learn how to use EasyMatch to learn a model from an ROI in a source image, and to perform pattern matching on the same image.

You'll need first to load the source image, and define an ROI inside (steps 1-2). Then, you'll have to learn the model, using this ROI (step 3). Finally, you'll perform pattern matching in the source image (step 4), and will find additional occurrences of the model.



ROI that will be learned



Occurrences matching the model ROI

Step 1: Load the source image

1. From the main menu, click **Image**, then **Open**.
2. Load the image file `EasyMatch\BOARD.JPG`.
3. Keep the default variable name for the new image object, and click **OK**.

Step 2: Define an ROI

1. Right-click in the image, and select **New ROI...** from the contextual menu.
2. Keep the default variable name for the new ROI object, and click **OK**. A default ROI is placed over the image (blue rectangle with handles).

The ROI management dialog box is opened.

3. Resize the ROI and move it around one of the blue capacitors at the lower left part of the image.

Step 3: Learn a model from the ROI

1. From the main menu, click **EasyMatch**, then **New Match Tool**.
2. Keep the default variable name for the new matcher object, and click **OK**.
3. In the **Learning** tab of the matcher dialog box, select the ROI object from the Source Image drop-down list, and click **Learn** to acquire the model pattern.

Step 4: Match the pattern

1. In the **Matching** tab, increase the **Max Occurrences** field to 2.
2. Select the image object from the Source Image drop-down list.
3. Click **Execute**. The occurrences of the learned model are highlighted in the source image.
4. Further information about each occurrence can be found by clicking **Results**.
5. Click in a row to see the corresponding occurrence highlighted in the image.

Improving the Score of Matching Instances by Using "Don't Care Areas"

Setting Search Parameters

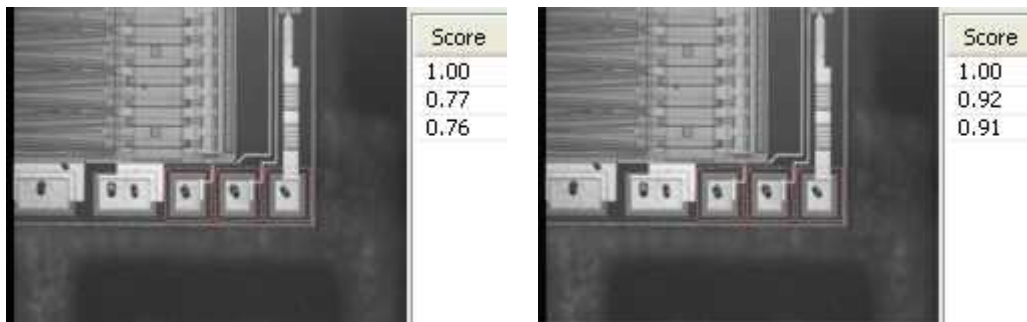
Objective

Following this tutorial, you will learn how to use EasyMatch to handle "don't care areas". "Don't care areas" help to define in the image the meaningful features only, by masking the areas that might change from image to image.

You'll need first to load a reference image and learn the reference model (steps 1-2). Then you'll perform automatic pattern matching of instances of the reference model, without using "don't care areas" (step 3). As the matching scores of the found instances are not high enough, you'll define a "don't care area" on the reference model, and restart the detection. The matching scores are much better (steps 4-5).



Reference model



Found instances and matching scores, without (left) and with (right) using "don't care areas"

Step 1: Load the reference image

1. From the main menu, click **EasyMatch**, then **New Match Tool**.
2. Keep the default variable name for the new Matcher object, and click **OK**. The Matcher management dialog box is opened.
3. In the **Learning** tab, click the **Open** icon, and load the image file EasyMatch\Die Pad Model 1.bmp.
4. Keep the default variable name for the new Image object, and click **OK**.

Step 2: Learn the reference model

- Click **Learn** to acquire the model pattern.

Step 3: Detect instances of the reference model without "don't care areas"

1. In the **Matching** tab, click the **Open** icon, and load the image file EasyMatch\Die Pad 1.bmp.
2. Keep the default variable name for the new Image object, and click **OK**. An instance matching the reference model is highlighted.
3. Increase the **Max Occurrences** field to 3. Enable the **Sub-Pixel Interpolate** check-box to increase the matching precision.
4. Enter '-10.0' as the Minimum Angle (Deg). (Check that the angle is displayed in degrees. If not, select the angles unit from **View > Option** menu.)
5. Enter '10.0' as the Maximum Angle (Deg).
6. Click **Execute**. The pattern locations are highlighted in the source image.
7. Click **Results** to see the matching score of each found instance. The last two scores are rather bad. This is mainly due to the bright rectangle on the upper part of the reference image we have learned. We can improve the scores by using a "don't care area" to mask this bright area.

Step 4: Define the "don't care area"

1. In the **Don't Care Areas** tab, select the **Rectangle** radio button from the **Blacken Inside** group.
2. In the reference image, move your mouse pointer on the lower left corner of the bright rectangle, left-click and drag the don't care area (rectangle with red stripes) to the upper right corner of the bright rectangle to mask out this area.
3. In the **Don't Care Areas** tab, click **Learn**.

Step 5: Detect instances of the reference model with "don't care areas"

1. In the **Matching** tab, click **Execute**. The instances matching the reference model are still highlighted.
2. Click **Results** to compare the new matching scores. They are much better.

2.6. EasyFind

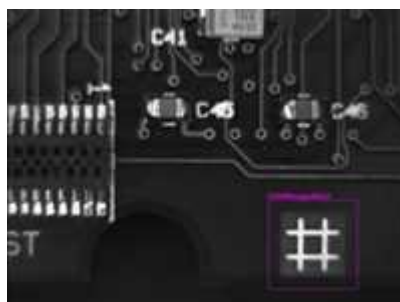
Detecting Highly-Degraded Occurrences of a Reference Model in Multiple Files

Pattern Finding and Retrieving Results

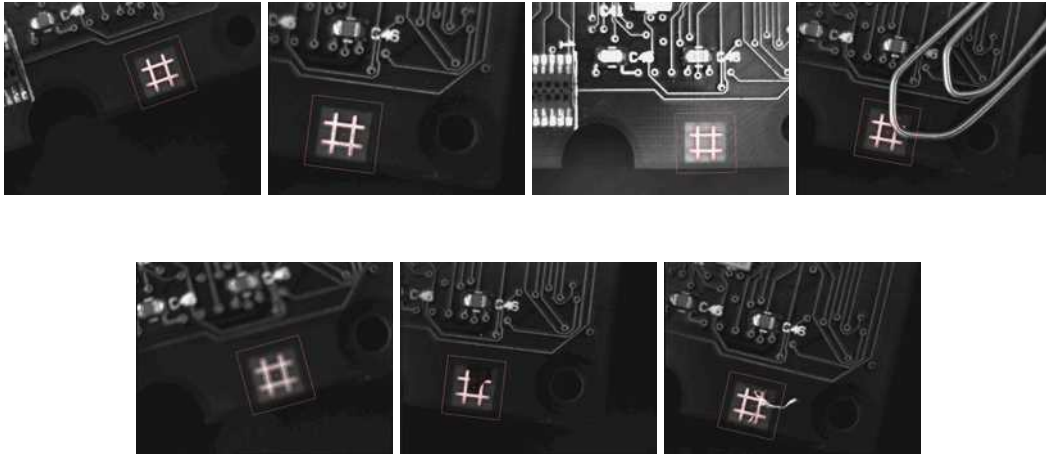
Objective

Following this tutorial, you will learn how to use EasyFind to detect in multiple images highly-degraded occurrences of a reference model. The degradation can be due to noise, blur, occlusion, missing parts or unstable illumination conditions.

You'll need first to load a reference image, define an ROI where EasyFind will learn the reference model, and set rotation and scaling tolerances for the expected occurrences to search for (steps 1-4). Then you're ready to open multiple files, and perform automatic detection of occurrences (even highly-degraded) of the reference model (steps 5-6).



Reference model



Occurrences of the reference model are found, even if highly-degraded

Step 1: Load the reference image

1. From the main menu, click **EasyFind**, then **New EasyFind Tool**.
2. Keep the default variable name for the new PatternFinder object, and click **OK**. The PatternFinder management dialog box is opened.
3. In the **Model** tab, click the **Open** icon, and load the image file `EasyFind\Fiducial 1.tif`.
4. Keep the default variable name for the new image object, and click **OK**.

Step 2: Create an ROI to define the reference model on the reference image

1. In the image, right-click and select **New ROI...** item from the menu.
2. Keep the default variable name for the new ROI object, and click **OK**. A default ROI is placed over the image (blue rectangle with handles). The ROI management dialog box is opened.
3. Drag the ROI over the reference model and resize it using its handles. Alternatively, enter the following coordinates in the ROI dialog box: 500, 365, 170, 170 for OrgX, OrgY, Width, and Height respectively, and click **Close**.

Step 3: Learn the reference model

1. In the PatternFinder **Model** tab, select the ROI object from the source image drop-down list, and click **Learn**. The reference model is perfectly detected (green edges).
2. In the PatternFinder **Search Field** tab, select the Image object from the source image drop-down list. Tick the Draw Features check-box.

The model location and feature points are highlighted in the source image.

Step 4: Set rotation and scaling tolerances

- In the PatternFinder **Allowances** tab, set both angle tolerance and scale tolerance to 25.0.

Step 5: Select multiple images

1. In the PatternFinder **Search Field** tab, click the **Open** icon. Select the images files EasyFind\Fiducial 2.tif to Fiducial 8.tif (use the shift key to select multiple files), and click **Open**.
2. Keep the default variable name for the new Image object, and click **OK**. The last image appears. The reference model is found, even if highly-degraded.
3. Detection of the reference model is automatically performed. It is not necessary to click Find once a new image appears, as inspection is automatically performed. However, clicking **Find** will insert the corresponding code into the script windows.
4. Click **Results** to find more information about the found instance.

Step 6: Browse multiple images

- In the PatternFinder **Search Field** tab, click the **Load Next** or **Load Previous** icons.

The image files appear, and the reference model is automatically detected.

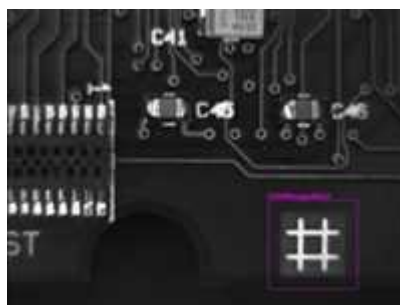
Improving the Score of Found Instances by Using "Don't Care Areas"

Setting Search Parameters

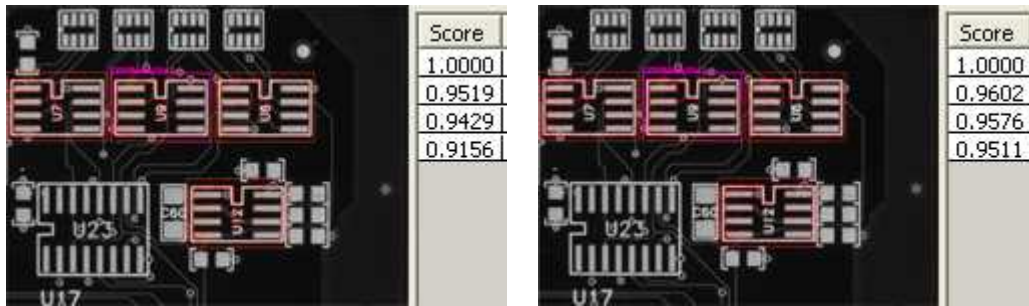
Objective

Following this tutorial, you will learn how to use EasyFind to handle "don't care areas" in geometric pattern matching. "Don't care areas" help to define in the image the meaningful features only, by masking the areas that might change from image to image, such as text and numbers.

You'll need first to load a reference image, define an ROI where EasyFind will learn the reference model, and set a rotation tolerance for the expected instances to search for (steps 1-4). Then you're ready to perform automatic detection of instances of the reference model, without using "don't care areas" (step 5). As the matching scores of the found instances are not high enough, you'll define a "don't care area" on the reference model, and restart the detection. The matching scores are slightly better (steps 6-7).



Reference model



Found instances and matching scores, without (left) and with (right) using "don't care areas"

Step 1: Loading the reference image

1. From the main menu, click **EasyFind**, then **New EasyFind Tool**.
2. Keep the default variable name for the new PatternFinder object, and click **OK**. The PatternFinder management dialog box is opened.
3. In the **Model** tab, click the **Open** icon, and load the image file EasyFind\Solder Pad 1.tif.
4. Keep the default variable name for the new Image object, and click **OK**.

Step 2: Creating an ROI to define the reference model on the reference image

1. In the image, right-click and select **New ROI...** item from the menu.
2. Keep the default variable name for the new ROI object, and click **OK**. A default ROI is placed over the image (blue rectangle with handles). The ROI management dialog box is opened.
3. Drag the ROI over the reference model and resize it using its handles. Alternatively, enter the following coordinates in the ROI dialog box: 200, 130, 190, 130 for OrgX, OrgY, Width, and Height respectively, and click **Close**.

Step 3: Learning the reference model

1. In the PatternFinder **Model** tab, select the ROI object from the source image drop-down list, and click **Learn**. The reference model is detected.
2. In the PatternFinder **Search Field** tab, select the Image object from the source image drop-down list. Tick the **Draw Features** check-box. The model location and feature points are highlighted in the source image.

Step 4: Setting a rotation tolerance

1. In the PatternFinder **Allowances** tab, set the angle tolerance to 5.0.

Step 5: Detecting instances of the reference model without "don't care areas"

1. In the PatternFinder **Search Field** tab, set **Max Instances** to 4, and click **Find**. The instances matching the reference model are highlighted.
2. Click **Results** to see the matching score of each found instance. Even though the scores are good, we can still improve them slightly by using a "don't care area" to mask the text appearing in the learned pattern.

Step 6: Defining the "don't care area"

1. In the PatternFinder **Don't Care Areas** tab, select the **Rectangle** radio button from the **Blacken Inside** group.
2. In the ROI defining the reference model, move your mouse pointer on the top-left corner of the text "U9", left-click and drag the don't care area (rectangle with red stripes) to mask out the text.

Step 7: Detecting instances of the reference model with "don't care areas"

1. In the PatternFinder **Don't Care Areas** tab, click **Learn**, and then click **Find**.

The instances matching the reference model are still highlighted, but the text is not found anymore.

2. Click **Results** to compare the new matching scores.

They are slightly better.

2.7. EasyGauge

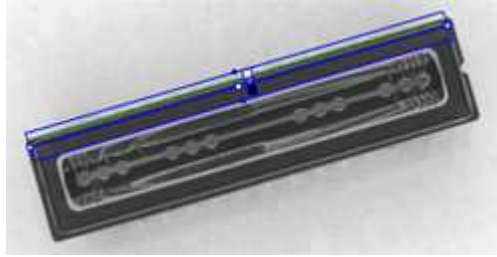
Measuring the Rotation Angle of an Object

Line Fitting

Objective

Following this tutorial, you will learn how to use EasyGauge to measure the rotation angle of a CCD sensor package. As we only need to retrieve an angle value, it's not required to work in a calibrated field of view. All geometrical parameters and results will be express as numbers of pixels.

You'll need to load the source image (step 1), and attach a line fitting gauge (step 2). The inspection is automatically performed (step 3).



Line fitting gauge

Step 1: Load the source image

1. From the main menu, click **EasyGauge**, then **New World Shape**.
2. Keep the default variable name, and click **OK**.
3. From the **Gauges** tab, click the **Open** icon, and load the image file `EasyImage\CCD.tif`.
4. Keep the default variable name for the new image object, and click **OK**.

Step 2: Attach a line gauge to the image

1. In the **Gauges** tab of the world shape dialog box, right-click the world shape icon, select **New > Line Gauge** from the contextual menu.
2. Keep the default variable name, and click **OK**. The line location gauge appears on the image. It consists of the following elements:
 - A blue line segment along which the transitions search is carried out.
 - Five white handles, allowing the user to move and rotate the segment.
 - A gray arrow, indicating in which direction the segment is traversed.
 - Black and white rectangles, indicating which kind of transition is searched for.
 - Green line, indicating the transition points found (if any).
3. Using the handles, move, rotate and extend the line gauge, so that it is positioned on the upper edge of the CCD package, with the gray arrow pointing downwards.
4. In the **Measurement** tab of the line gauge dialog box, choose 'White to Black' from the Type dropdown list and 'From Begin' from the Choice dropdown list.

Step 3: Perform the inspection

1. The image is automatically inspected. However, clicking **Process** in the world shape dialog box will insert the corresponding code into the script window.
2. Click the **Results** tab to retrieve the measured angle value.
3. To see the individual fitting points, select the **Points** checkbox under the Draw Samples area.

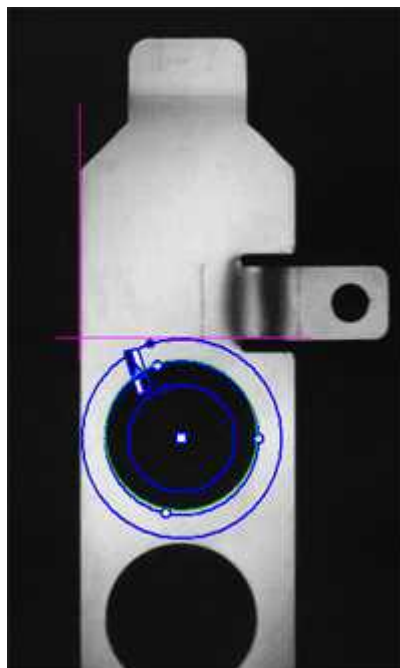
Measuring the Diameter of a Circle

Circle Fitting

Objective

Following this tutorial, you will learn how to use EasyGauge to measure the diameter of a circle in an image.

You'll first load an image for calibration —a dot grid— (step 1), and calibrate the field of view (step 2). Then you'll load the source image for inspection (step 3), and attach a circle fitting gauge (step 4). The inspection is automatically performed (step 5). All measurement results are expressed in physical units..



Measuring the diameter of a circle

Step 1: Load the calibration image

1. From the main menu, click **EasyGauge**, then **New World Shape**.
2. Keep the default variable name, and click **OK**.
3. In the **Dot Grid Calibration** tab, click the **Open** icon, and load the image file `EasyGauge\Dot Grid 1.tif`.
4. Keep the default variable name for the new image object, and click **OK**.

Step 2: Calibrate the field of view

- Click **Calibrate**. From now on, the field of view is calibrated, and all results will be expressed in physical units.

Step 3: Load the source image

1. From the **Gauges** tab, click the Open icon, and load the image file EasyGauge\Bracket 6.tif.
2. Keep the default variable name, and click **OK**.

Step 4: Attach a circle gauge to the image

1. In the **Gauges** tab of the world shape dialog box, right-click the frame shape icon, select **New > Circle Gauge** from the contextual menu.
2. Keep the default variable name, and click **OK**.
3. The circle location gauge appears on the image. It consists of the following elements:
 - A blue ring in which the circle is searched for.
 - A blue nominal circle.
 - Six white handles, allowing the user to move and rotate the segment.
 - A gray arrow, indicating in which direction the segment is traversed.
 - Black and white rectangles, indicating which kind of transition are searched for.
 - A green arc of circle, indicating the circle found (if any).
4. Using the handles, drag the circle fitting gauge around the upper bracket hole. Adjust the nominal circle on the hole edge and extend the searching area if necessary.
5. In the **Measurement** tab of the circle gauge dialog box, select 'From Begin' from the Choice dropdown list.

Step 5: Perform the inspection

1. The image is automatically inspected. However, clicking **Process** in the world shape dialog box will insert the corresponding code into the script window.
2. Click the **Results** tab to retrieve the measured diameter. All measurement results are expressed in physical units.

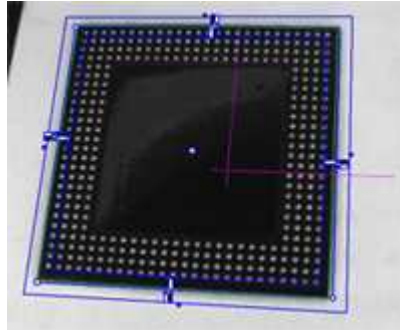
Measuring a Distorted Rectangle

Rectangle Fitting

Objective

Following this tutorial, you will learn how to use EasyGauge to perform measurements on a distorted rectangle component.

To obtain measurement results in physical units, we need to work in a calibrated field of view. You'll need first to load an image for calibration —a dot grid— (step 1), and calibrate the field of view (step 2). Then you'll load the distorted image (step 3), and attach a rectangle fitting gauge (step 4). The inspection is automatically performed (step 5). All measurement results are expressed in physical units.



Measuring a distorted rectangle

Step 1: Load the calibration image

1. From the main menu, click **EasyGauge**, then **New World Shape**.
2. Keep the default variable name, and click **OK**.
3. In the **Dot Grid Calibration** tab, click the **Open** icon, and load the image file `EasyGauge\Dot Grid 5.tif`. This dot grid has been acquired in the same conditions and has the same distortion as the image we want to inspect.
4. Keep the default variable name for the new image object, and click **OK**.

Step 2: Calibrate the field of view

- Click **Calibrate**. From now on, the field of view is calibrated, and all results will be expressed in physical units.

Step 3: Load the distorted image

1. From the **Gauges** tab, click the **Open** icon, and load the image file `EasyGauge\Distorted Component.tif`.
2. Keep the default variable name, and click **OK**.

Step 4: Attach a rectangle gauge to the image

1. In the **Gauges** tab, right-click the world shape icon, and select **New > Rectangle Gauge** from the contextual menu.
2. Keep the default variable name, and click **OK**. The rectangle gauge dialog box is opened, and the rectangle gauge is drawn on the image. It consists of the following elements:
 - A blue rectangular ring in which the rectangle is searched for.
 - A blue nominal rectangle.

- Eleven white handles, allowing the user to move and extend the search area.
 - Gray arrows, indicating in which direction segments are examined.
 - Black and white rectangles, indicating which kind of transition are searched for.
 - A green rectangle, indicating the rectangle found (if any).
3. Due to the perspective effect, the rectangle gauge doesn't look like a rectangle. Using the central handle, move the rectangle gauge in the image and observe the rectangle deformation. Due to the calibration, the rectangle gauge shape adapts to the field of view deformation. Extend the searching area, and adjust the nominal rectangle on the component edges.
 4. In the **Measurement** tab of the rectangle gauge dialog box, select 'White To Black' from the Type dropdown list and 'From Begin' from the Choice dropdown list.

Step 5: Perform the inspection

1. The image is automatically inspected. However, clicking **Process** in the world shape dialog box will insert the corresponding code into the script window.
2. Click the **Results** tab to retrieve the measured X and Y sizes. All measurement results are expressed in physical units.

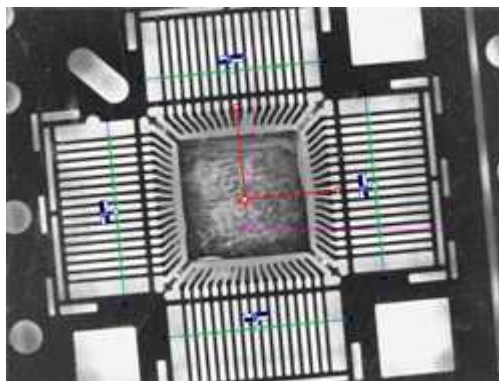
Locating Points Regarding to a Coordinate System

Point Location

Objective

Following this tutorial, you will learn how to use EasyGauge to perform lead frames inspection. This operation determines the dimension, position, curvature, size, angle or diameter of the lead frames with an excellent accuracy. Robustness is ensured by powerful edge-point selection mechanisms that are intuitive and easy to tune, allowing measurement in cluttered images.

You'll first load an image for calibration —a dot grid— (step 1), and calibrate the field of view (step 2). Then you'll load the lead frame image (step 3), and set a coordinate system (a frame shape). Regarding to this coordinate system, you can define point gauges (steps 5-6). Finally, you'll load another lead frame image, that has a slight angle deviation, so the coordinate system has to be rotated (steps 7-8). The inspection is then automatically performed (step 9). All measurement results are expressed in physical units.



Four point gauges over four sets of leads

Step 1: Load the calibration image

1. From the main menu, click **EasyGauge**, then **New World Shape**.
2. Keep the default variable name, and click **OK**.
3. In the **Dot Grid Calibration** tab, click the **Open** icon, and load the image file `EasyGauge\Dot Grid 2.tif`.
4. Keep the default variable name for the new image object, and click **OK**.

Step 2: Calibrating the field of view

1. Click **Calibrate**. From now on, the field of view is calibrated, and all results will be expressed in physical units.

Step 3: Loading a lead frame image

1. From the **Gauges** tab, click the **Open** icon, and load the image file `EasyGauge\Lead Frame 1.tif`.
2. Keep the default variable name, and click **OK**.

Step 4: Setting a coordinate system

1. In the **Gauges** tab of the world shape dialog box, right-click the world shape icon, select **New > Frame Shape** from the contextual menu.
2. Keep the default variable name, and click **OK**. The frame shape icon appears in the world shape dialog box.
3. Drag the frame shape center approximately to the square center of the image.

Step 5: Attaching a point gauge to the frame shape

1. In the **Gauges** tab of the world shape dialog box, right-click the frame shape icon, select **New > Point Gauge** from the contextual menu.
2. Keep the default variable name, and click **OK**. The point location gauge appears on the image. It consists of the following elements:
 - A blue line segment along which the transitions search is carried out.
 - Three white handles, allowing the user to move and rotate the segment.
 - A gray arrow, indicating in which direction the segment is traversed.
 - Black and white rectangles, indicating which kind of transition is searched for.
 - Green crosses, indicating the transition points found (if any).
3. Place the point location gauge over a set of leads: in the **Position** tab of the point gauge dialog box, set the Center Y property to 7, and the Tolerance property to 5.

Step 6: Attaching other point gauges to the frame shape

1. From the **Gauges** tab of the world shape dialog box, create three other point gauges (refer to step 5).
2. Place these point location gauges over the remaining sets of leads :
 - Center Y = -7, Tolerance = 5
 - Center X = 7, Tolerance = 5, Angle = 90
 - Center X = -7, Tolerance = 5, Angle = 90

Step 7: Loading another lead frame image

1. From the **Gauges** tab, click the Open icon, and load the image file EasyGauge\Lead Frame 2.tif.
2. Keep the default variable name for the new image, and click **OK**.

Step 8: Tuning the coordinate system

1. In the frame shape dialog box, set the Angle property to 5.8.
2. Drag the frame shape center approximately to the square center of the image. All point location gauges automatically follow.

Step 9: Performing the inspection

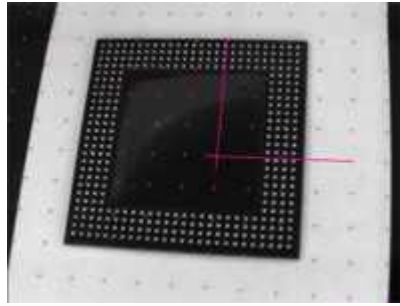
1. The image is automatically inspected. However, clicking Process in the world shape dialog box will insert the corresponding code into the script window.
2. From the point gauge dialog boxes, click the Results tab to retrieve the located points coordinates. All measurement results are expressed in physical units. The values refer to the frame shape system.

Unwarping a Distorted Image

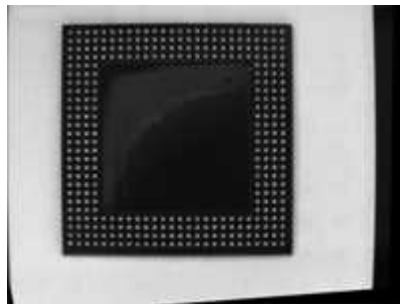
Objective

Following this tutorial, you will learn how to use EasyGauge to perform grid calibration, and unwarped a distorted image.

You'll first load an image for calibration—a dot grid— (step 1), and calibrate the field of view (step 2). Then you'll load the distorted image (step 3), and perform the unwarping operation (step 4).



Distorted image



Unwarped image

Step 1: Load the calibration image

1. From the main menu, click **EasyGauge**, then **New World Shape**.
2. Keep the default variable name, and click **OK**.
3. In the **Dot Grid Calibration** tab, click the **Open** icon, and load the image file `EasyGauge\Dot Grid 5.tif`. This dot grid has been acquired in the same conditions and has the same distortion as the image we want to unwarped.
4. Keep the default variable name for the new image object, and click **OK**.

Step 2: Calibrate the field of view

- Click **Calibrate**.

Step 3: Load the distorted image

1. From the **Unwarping** tab, click the **Open** icon, and load the image file `EasyGauge\Distorted component.tif`.
2. Keep the default variable name, and click **OK**.

Step 4: Unwarp the distorted image

1. In the Destination Image area, click **New** icon to create a new image.
2. Keep the default image settings, and click **OK**.

3. Select **Interpolate** checkbox to improve resulting image quality.
4. Click **Unwarp**. In the destination image, all distortions are corrected.

2.8. EasyOCR

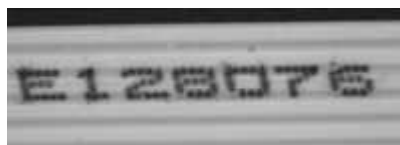
Learning Characters and Creating an EasyOCR Font

Learning Characters

Objective

Following this tutorial, you will learn how to use EasyOCR to learn new characters and save them in an EasyOCR font.

You'll need first to load a source image (step 1). Then you'll set the segmentation parameters to isolate each character (step 2). Each character will have to be learnt (step 3), and finally you'll save all the learnt characters as a font file (step 4). You can also add new characters to an existing font if needed (step 5).



Source image



The image is segmented so that all the characters are detected



All the characters have been learn

Step 1: Load the source image

1. From the main menu, click **EasyOCR**, then **New OCR Tool**.

2. Keep the default variable name for the new OCR object, and click **OK**.
3. In the **Source Image** tab, click the **Open** icon of the Source Image area, and load the image file EasyOCR\FlatCable\FlatCable1.tif.
4. Keep the default variable name for the new image object, and click **OK**.

Step 2: Set segmentation parameters

1. Select the **Segmentation Parameters** tab, and move the red frame in the image above a character.
2. Tune each property to get a green bounding box around each character:
 - threshold value** = 113
 - characters** color = Black on White
 - min width** = 36
 - min height** = 31
 - spacing** = 4
 - max width** = 98
 - max height** = 72
 - noise area** = 9

Step 3: Learn new characters

1. Select the **Learn** tab, and click the character E in the image. You are then prompted to identify the character along with its class. Enter E in the character field, and select the 'EOcrClass_Uppercase' class. Click **OK**. Whenever a character has been added to the current font, its bounding box turns yellow.
2. Click the character 1 in the image. Enter 1 in the character field, and select the 'EOcrClass_Digit' class. Click **OK**.
3. Proceed with remaining characters.

Step 4: Save the EasyOCR font

- In the **Font File** tab, click the **Save As...** button. Type a file name for the new EasyOCR font file. Its extension will be .ocr. Finally, click **Save**.

Step 5: Add characters to an existing font

1. In the Source Image tab, click the **Open** icon of the Source Image area, and load the image file EasyOCR\FlatCable\FlatCable2.tif.
2. Keep the default variable name for the new image object, and click **OK**.
3. In the **Recognition** tab, click **Execute**. Characters 2 and 8 are read correctly, but A, W and G are not (**low confidence score**). They don't belong to the font.
4. Select the **Learn** tab, and learn the characters A, W, and G (refer to step 3).

- Then save the font again (refer to step 4). The new characters have been added.

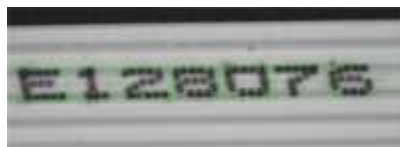
Recognizing Characters According to a Font

"Recognizing Characters According to a Font" 위

Objective

Following this tutorial, you will learn how to use EasyOCR to recognize characters, regarding to a specific font.

You'll need first to load a source image (step 1), and an EasyOCR font file (step 2). Then you'll perform the characters recognition (step 3).



Characters matching the font are automatically detected



Results after explicit recognition

Step 1: Load the source image

- From the main menu, click **EasyOCR**, then **New OCR Tool**.
- Keep the default variable name for the new OCR object, and click **OK**.
- In the Source Image tab, click the **Open** icon of the Source Image area, and load the image file EasyOCR\FlatCable\FlatCable1.tif.
- Keep the default variable name for the new 위 image object, and click **OK**.

Step 2: Load the font file

- In the **Font File** tab, click **Load**, and select the font file EasyOCR\FlatCable\FlatCable.ocr. In the image, the detected characters are highlighted in green.

Step 3: Recognize the characters

- In the **Recognition** tab, click **Execute** to trigger the recognition of the detected characters. The recognized characters appear in the Recognition results area. Further information about each character can be found in the table.

2.9. EasyOCV

Creating an EasyOCV Model File

Creating an OCV Model

Objective

Following this tutorial, you will learn how to build an EasyOCV model file.

You'll need first to load an image (step 1). Then you'll learn the model file (step 2), and perform the mark inspection (step 3).



Learned model

Step 1: Load the source image

1. From the main menu, click **EasyOCV**, then **New OCV Tool**.
2. Keep the default variable name, and click **OK**.
3. In the **Learn** tab, click the **Open** icon of the Source Image area, and load the image file EasyOCV\T20\T20 A.bmp.
4. Keep the default variable name, and click **OK**.

Step 2: Learn the model

1. In the **Learn** tab, click **Next** subsequently for three times.
2. Click **Finish**.

The learning process is complete.

Step 3: Save the model file

1. In the **Inspect** tab, click the **Save As...** button.
2. Type a file name for the new EasyOCV model file. Its extension will be .ocv.
3. Finally, click **Save**.

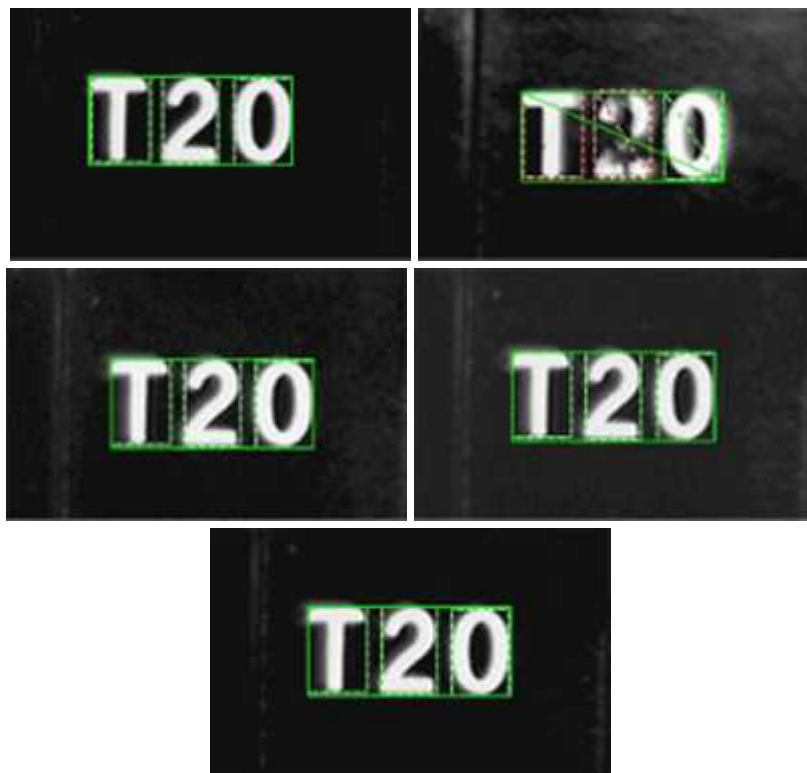
Inspecting Characters in an Image According to a Model File

Inspecting

Objective

Following this tutorial, you will learn how to load an EasyOCV model file, and to perform mark inspection in an image.

You'll need first to load source images (step 1), and a model file (step 2). Then, you'll perform the mark inspection (step 3).



Mark inspection in multiple source images

Step 1: Load the source images

1. From the main menu, click **EasyOCV**, then **New OCV Tool**.
2. Keep the default variable name for the new OCV object, and click **OK**.
3. In the **Inspect** tab, click the **Open** icon of the **Source Image** area, and load the image files `EasyOCV\T20\T20 A.bmp` to `T20 E.bmp`. Use the shift key to select multiple files.
4. Keep the default variable name for the new image object, and click **OK**.

Step 2: Load the model file

1. In the **Inspect** tab, click the **Load** button.
2. Select the model file `EasyOCV\T20\T20.ocv`.

Step 3: Perform the mark inspection

1. In the **Inspect** tab, click **Execute** to trigger the mark inspection.
Texts and characters are highlighted in the source image.
2. In the **Parameters and results** tab, click **Character Overview** and **Text Overview** to display further information on characters and texts.
3. Click the **Quality** button, and select characters in the image, to display quality indicators.
4. In the **Inspect** tab, click the **Load Next** and **Load Previous** icons to browse through the multiple images.

In the `T20 E` image, the character 2 is underprinted, and the character 0 is overprinted.

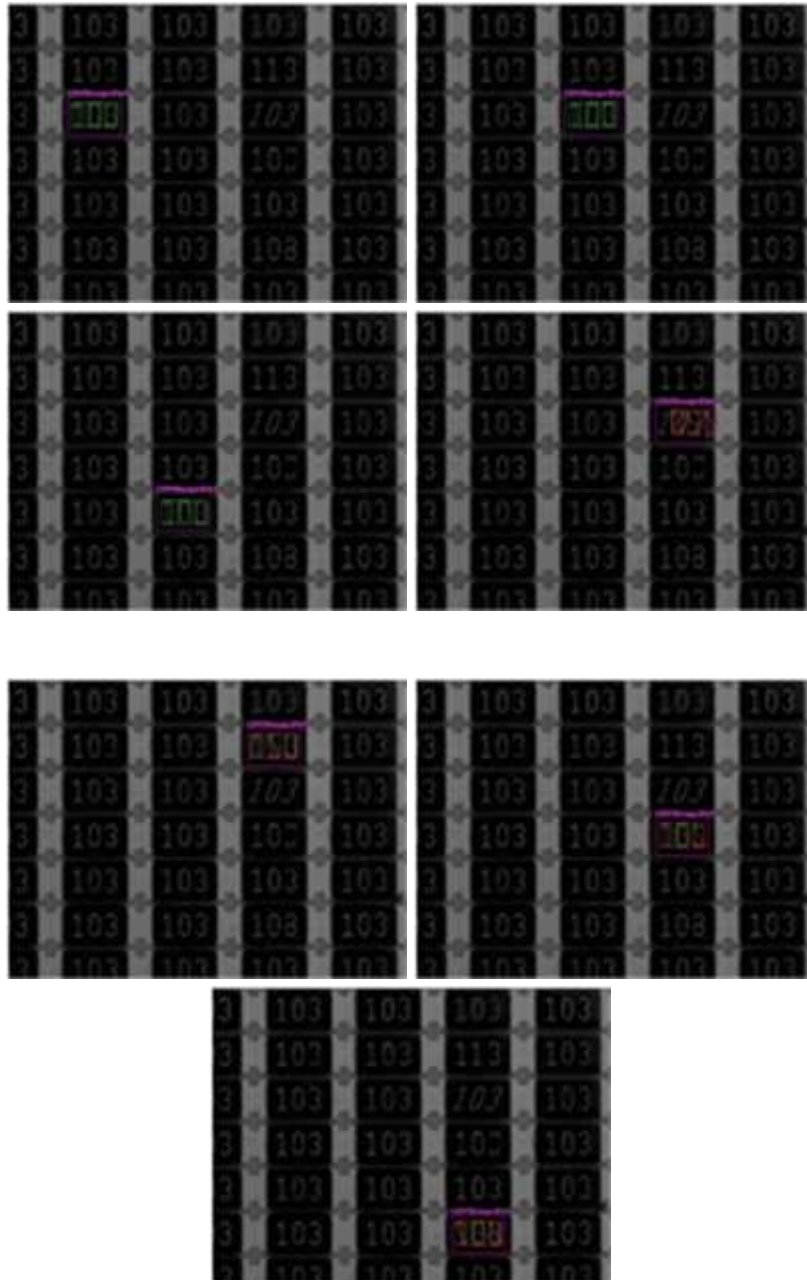
Inspecting Characters in an ROI According to a Model File

Inspecting

Objective

Following this tutorial, you will learn how to load an EasyOCV model file, and to perform mark inspection in an ROI of a source image.

You'll need first to load a source image and define an ROI where to perform the mark inspection (steps 1-2). Then, you'll load a model file (step 3). Finally, you'll perform the mark inspection (step 4).



Mark inspection in multiple ROIs

Step 1: Load the source image

1. From the main menu, click **EasyOCV**, then **New OCV Tool**.
2. Keep the default variable name for the new OCV object, and click **OK**.
3. In the **Inspect** tab, click the **Open** icon of the Source Image area, and load the image file `EasyOCV\103\103.bmp`.
4. Keep the default variable name for the new image object, and click **OK**.

Step 2: Define the ROI

1. Right-click in the image, and select **New ROI...** from the contextual menu.
2. Keep the default variable name for the new ROI object, and click **OK**.

A default ROI is placed over the image (blue rectangle with handles). The ROI management dialog box is opened.

3. Enter the following coordinates in the ROI management dialog box: 118, 181, 104, 74 for OrgX, OrgY, Width, and Height respectively.

Step 3: Load the model file

1. In the **Inspect** tab, click the **Load** button.
2. Select the model file `EasyOCV\103\103.ocv`.

Step 4: Perform the mark inspection

1. In the **Inspect** tab, select the ROI object from the Source Image drop-down list, and click **Execute** to trigger the mark inspection. Texts and characters are highlighted in the source image.
2. In the **Parameters and results** tab, click **Character Overview** and **Text Overview** to display further information on characters and texts.
3. Click the **Quality** button, and select characters in the image, to display quality indicators.
4. Change the ROI position (OrgX, OrgY), and click **Execute Inspection** at each new position to trigger the inspection again:
 - (288, 181): OK.
 - (288, 349): OK.
 - (462, 181): the numbers are slanted.
 - (462, 95): the number 0 is wrong.
 - (462, 266): the number 3 is broken.
 - (462, 438): the number 3 is out of shape.

Learning a Model Using Statistics (1)

Statistical Learning

Objective

Following this tutorial, you will learn how to use EasyOCV to learn an EasyOCV model, and tune it using statistics.

You'll need first to load a reference image (step 1), and learn the model (step 2). Then you'll add tolerances on the model position (step 3), and use statistics on several images to tune the model (step 4). Finally, you'll perform the mark inspection on multiple images (step 5).

EasyOCVISE



Mark inspection in multiple images

Step 1: Load the reference image

1. From the main menu, click **EasyOCV**, then **New OCV Tool**.
2. Keep the default variable name, and click **OK**.

3. In the **Learn** tab, click the **Open** icon of the Source Image area, and load the image file EasyOCV\ISE\ISE 01.bmp.
4. Keep the default variable name, and click **OK**.

Step 2: Learn the model

1. In the **Learn** tab, click **Next** to edit the segmented objects. The characters "ASS", "MB" and "an" are grouped (same color segmentation). They need to be separated.
2. In the image, draw lines to separate touching characters.
3. Click **Next** twice, and click **Finish**.

The learning process is complete.

Step 3: Add tolerances to the model

1. In the **Parameters and results** tab, click **Position** in the **Selected texts parameters** area.
2. To allow more translation freedom, enter 80 and 60 as ShiftX and ShiftY tolerances.
3. Click **Close**. At this stage, the building of the basic EasyOCV model is achieved.

Step 4: Use statistics to tune the model

1. In the **Inspect** tab, click the **Open** icon of the Source Image area, and load the image files EasyOCV\ISE\ISE 01.bmp to ISE 05.bmp. Use the shift key to select multiple files.
2. Keep the default variable name and click **OK**.
3. In the **Inspect** tab, click **Add to statistics**. Click the **Load Next** icon, and click **Add to statistics**, and so on for all the images.
4. In the **Parameters and results** tab, click the **Quality** button of the **Statistic Tolerance Adjustment** area.
5. Keep the default value in the prompt box. Click **OK**.

Step 5: Inspect multiple images

1. In the **Inspect** tab, click the **Open** icon of the Source Image area, and load the image files EasyOCV\ISE\ISE 01.bmp to ISE 14 (bad character).bmp. Use the shift key to select multiple files.
2. Keep the default variable name for the new image object, and click **OK**.
3. In the **Inspect** tab, click the **Load Next** and **Load Previous** icons to browse through the multiple images. Take note of the following observations and results.
 - The images ISE 01 to ISE 05 have no defects.
 - In the image ISE 06, the text PBGA is badly printed.
 - In the image ISE 07, the mark is distorted.
 - In the image ISE 08, the mark is blurred.
 - In the image ISE 09, the characters 5 and 2 are incorrect.

- In the image ISE 10, characters are badly shaped.
- In the image ISE 11, characters 388 are underprinted.
- In the image ISE 12, characters ISE are overprinted.
- In the image ISE 13, two characters are missing.
- In the image ISE 14, a character is broken.

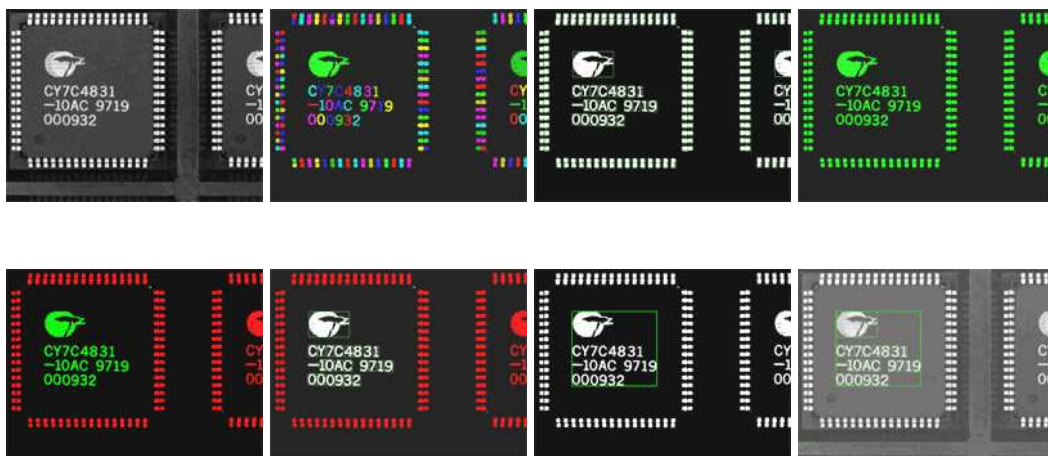
Learning a Model Using Statistics (2)

Statistical Learning

Objective

Following this tutorial, you will learn how to use the statistic feature and rotation tolerance of EasyOCV to build a model file.

You'll need first to load a reference image (step 1), and learn the model (text and logo) (step 2). Then you'll add tolerances on the model position (step 3), and use statistics on several images to tune the model (step 4). Finally, you'll perform the mark inspection on multiple images (step 5).



Learning text and logo

Step 1: Load the reference image

1. From the main menu, click **EasyOCV**, then **New OCV Tool**.
2. Keep the default variable name, and click **OK**.
3. In the **Learn** tab, click **Open** icon of the source image area, and load the image file `EasyOCV\CY7C\CY7C_01.jpg`.
4. Keep the default variable name, and click **OK**.

Step 2: Learn the model

1. In the **Learn** tab, click **Next** twice, the characters building dialog box appears.
2. Click **Undo Char(s)**. The text, logo and pins are now highlighted in green.

3. Using your mouse cursor, click-and-drag the rectangle over the text and the logo on the component.
4. Click **Form Auto Chars**.
5. Click **Next**, and then **Finish**. The learning process is complete.

Step 3: Add tolerances to the model

1. In the **Parameters and results** tab, click **Position** in the **Selected texts parameters** area.
2. To allow more translation freedom, enter 90 and 80 as ShiftX and ShiftY tolerances.
3. To allow translation and rotation, enter 5 as Skew tolerance, and 11 as Skew stride/count.
4. To speed up the location process, enter 7 as ShiftX and ShiftY stride/count.
5. Click **Close**. At this stage, building of the basic OCV model is achieved.

Step 4: Use statistics to tune the model

1. In the **Inspect** tab, click the **Open** icon of the Source Image area, and load the image files EasyOCV\CY7C\CY7C_01.jpg to CY7C_15.jpg. Use the shift key to select multiple files.
2. Keep the default variable name, and click **OK**.
3. Click **Add to statistics**. Click the **Load Next** icon, and click **Add to statistics**, and so on for all the images.
4. In the **Parameters and results** tab, click the **Quality** button from the Statistic Tolerance Adjustment area.
5. Set the Security Factor to 3.5, and click **OK**.

Step 5: Inspect multiple images

1. In the **Inspect** tab, click the **Open** icon of the Source Image area, and load the image files EasyOCV\CY7C\CY7C_01.jpg to CY7C_27.jpg. Use the shift key to select multiple files.
2. Keep the default variable name for the new image object, and click **OK**.
3. In the **Inspect** tab, click the **Load Next** and **Load Previous** icons to browse through the multiple images. Observe the detected defaults.

2.10. EasyBarcode

Reading Bar Codes Automatically

Reading a Bar Code

Objective

Following this tutorial, you will learn how to perform automatic reading of multiple bar codes. You'll need first to load multiple source images (step 1). The reading is then automatically performed on each image (step 2).



Each bar code is automatically detected and decoded

Step 1: Load the source images

1. From the main menu, click **EasyBarCode**, then **New BarCode Tool**.
2. Keep the default variable name, and click **OK**.
3. In the **AutoRead** tab, click the **Open** icon of the Source Image area, and load the image files `EasyBarCode\Barcode 01.tif` to `Barcode 10.tif`. Use the shift key to select multiple files.
4. Keep the default variable name, and click **OK**. The last image appears.

Step 2: Read the bar codes automatically

1. The bar code is automatically detected and decoded. The graphic result appears on the image, while the data content and the corresponding symbology are displayed in the Decoded Symbology area. It is not necessary to click **Read** once a new image appears. However, clicking **Read** will insert the corresponding code into the script window.

2. In the **Results** tab, find more information about the bar code. As a bar code might have a meaning under different symbologies, all possible contents are listed by decreasing likelihood.
3. In the **AutoRead** tab, click the **Load Next** and **Load Previous** icons to browse through images 01 to 06.

The image files appear, and each bar code is automatically detected and decoded. The bar code properties are updated.

4. To decode the remaining bar codes, we have to enable the additional symbologies.
5. In the **Symbologies** tab, click the **Toggle All** button of the Additional area.
6. In the **AutoRead** tab, click the **Load Next** and **Load Previous** icons and browse the remaining images.

2.11. EasyMatrixCode

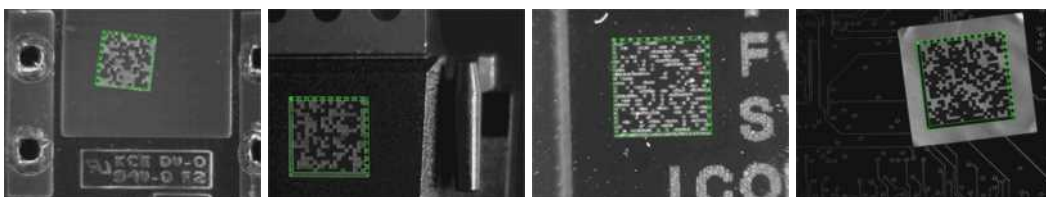
Reading Data Matrix Codes Automatically

Automatic Reading

Objective

Following this tutorial, you will learn how to use EasyMatrixCode to detect and decode automatically Data Matrix codes in multiple files.

You'll need first to load multiple source images (step 1). The reading is then automatically performed on each image (step 2). You can also grade the printing quality of each matrix code (step 3).



Each Data Matrix code is automatically detected and decoded

Step 1: Load the source images

1. From the main menu, click **EasyMatrixCode**, then **New MatrixCode Tool**.
2. Keep the default variable name for the new matrix code reader, and click **OK**.
3. In the **Read** tab, click the **Open** icon of the Source Image area, and load the image files EasyMatrixCode\AutoRead\AutoRead 01.tif to AutoRead 04.tif. Use the shift key to select multiple files.
4. Keep the default variable name for the new image, and click **OK**. The last image appears.

Step 2: Read the Data Matrix codes automatically

1. The Data Matrix code is automatically detected and decoded. The matrix code reference corner is highlighted with a bold cross mark. It is not necessary to click **Read** once a new image appears. However, clicking **Read** will insert the corresponding code into the script window.
2. In the **Results** area, find more information about the matrix code, such as the decoded string.
3. In the **Read** tab, click the **Load Next** and **Load Previous** icons. The image files appear, and each Data Matrix code is automatically detected and decoded. The matrix code properties are updated. If no Data Matrix code could be located in the image, an error message is displayed in the Message field.

Step 3: Grade Data Matrix code printing quality

1. In the **Print Quality** tab, select the **Compute Grading** check-box.
2. Click **Apply**.

For each printing quality parameter, the corresponding value and its grade equivalent appear. An A grade means a good quality, while an F grade indicates a poor one.

Learning a Data Matrix Code and Creating an EasyMatrixCode Model File

Reading with Prior Learning

Objective

Following this tutorial, you will learn how to use EasyMatrixCode to learn a Data Matrix code, and save it as an EasyMatrixCode model file.

You'll need first to load a source image (step 1), and learn the matrix code (step 2). Then you'll save the learned matrix code as a new model file (step 3). You can also add new learned matrix codes to an existing model if needed (step 4).



The Data Matrix code has been learned

Step 1: Load the source image

1. From the main menu, click **EasyMatrixCode**, then **New MatrixCode Tool**.
2. Keep the default variable name for the new matrix code reader object, and click **OK**.
3. In the **Learn** tab, click the **Open** icon of the Source Image area, and load the image file `EasyMatrixCode\Label\Label 4.tif`.
4. Keep the default variable name for the new image object, and click **OK**.

Step 2: Learn the Data Matrix code

- In the **Learn** tab, click **Learn**.

The Data Matrix code is detected and decoded without error.

The graphical result appears on the image.

The properties of the learned matrix code are updated in the dialog box.

Step 3: Save the model file

1. In the **Learn** tab, click the **Save As...** button.
2. Type a file name for the new EasyMatrixCode model file. Its extension will be `.mx2`.
3. Click **Save**.

Step 4: Learning more Data Matrix codes

1. In the **Read** tab, click the **Open** icon of the source image area, and load the image file `EasyMatrixCode\PCB Code\PCB Code 3.jpg`.
2. Keep the default variable name for the new image object, and click **OK**.
3. An error message is displayed in the Message area of the **Read** tab. The matrix code can not be read, since the reader uses the model from the "Label 4" image. You need to learn the "PCB Code 3" matrix code, and add it to the model.
4. In the **Learn** tab, click **Learn More**. The Data Matrix code is detected and decoded without error. The graphical result appears on the image. The properties of the learned matrix code are updated in the dialog box.
5. Using **Learn More** rather than **Learn** involves that the "Label 4" model is not replaced by the "PCB Code 3" model, but both are now included in the same model. In the **Read** tab, the "PCB Code 3" matrix code is correctly read. Select the "Label 4" image in the drop-down list of the source image area. The "Label 4" matrix code is still read without error, which means that both learned matrix codes have been kept.
6. Finally, save the model again (refer to step 3). The new matrix code has been added.