Image processing Project

Face detection & recognization

2nd ECE

Dr / Azhar Ahmed

Presented by our Team

Esraa Elshafey
Heba Magdy
Yasmin Ammar

Samar Nafea
Menna Mohamed
Yasmin Reda

Ziad Wael
Ziad Mahmoud
Ahmed Khaled

introduction

MATLAB-based face recognition application with modern dark-themed UI

Complete workflow from face detection to identity verification

Self-contained system requiring no external libraries

Manual Object Detection

System Architecture & Workflow

01

Image:

- Acquisition: Upload via file selection dialog

02

Preprocessing:

- Convert to gray image.
- apply histogram Equalization.
- smoothing.

03

Face Detection:

- Multi-stage approach using Using Function dedicated to extract face features like: eyes, skin and edge.

04

Feature Extraction:

- Multi-descriptor approach for robust representation

05

Face Database:

- Local storage of registered identities and feature vectors

06

Recognition System:

- Similarity scoring and threshold-based identification

Key Technical Components

01

Advanced Face Detection Pipeline

02

Multi-Modal Feature Extraction

03

Recognition Algorithm

04

UI/UX Implementation

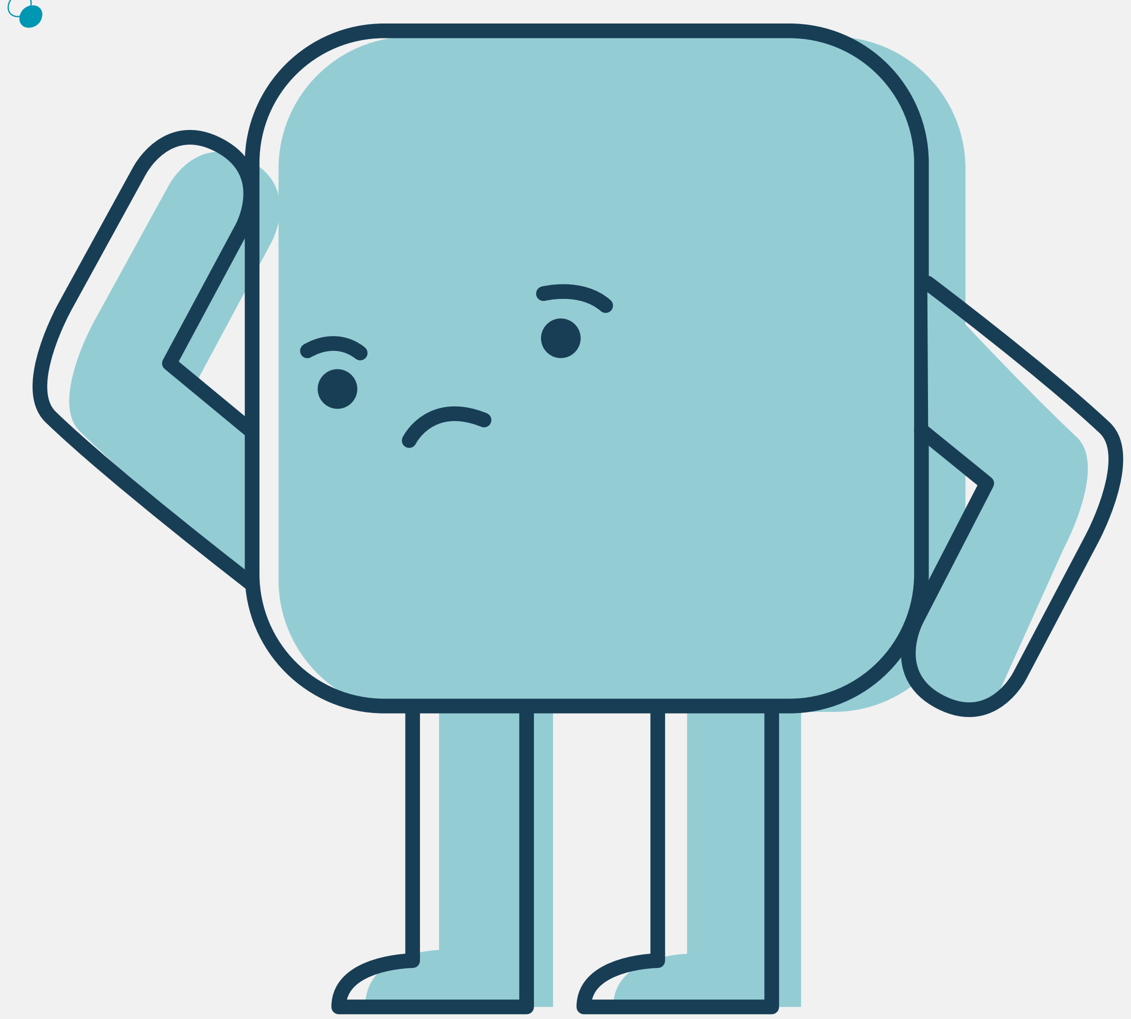
05

Performance Highlights

06

Implementation Details

What about our main functions in matlab !



Here We GOOO 🤞

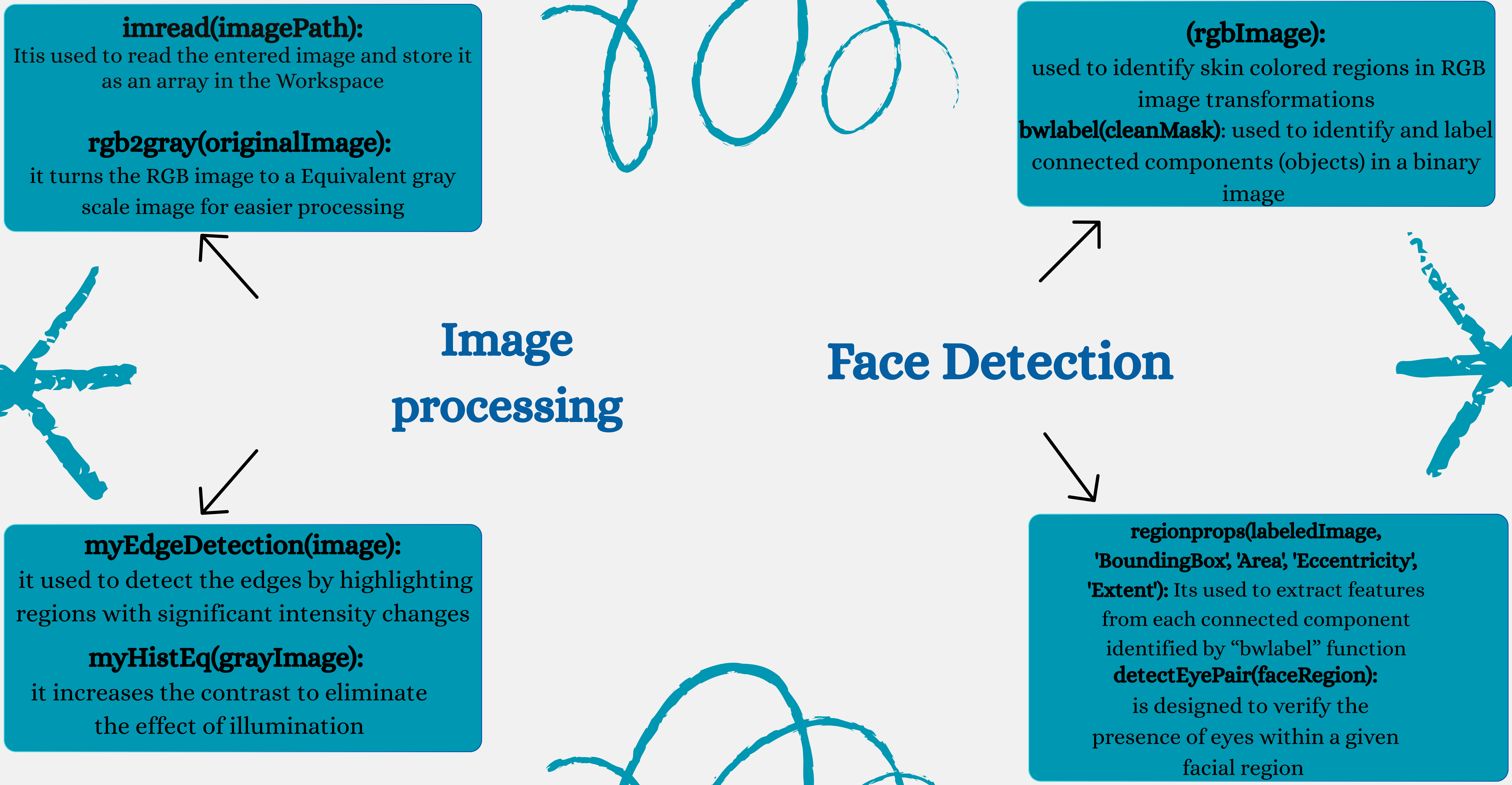


Image processing

imread(imagePath):
Itis used to read the entered image and store it as an array in the Workspace

rgb2gray(originalImage):
it turns the RGB image to a Equivalent gray scale image for easier processing

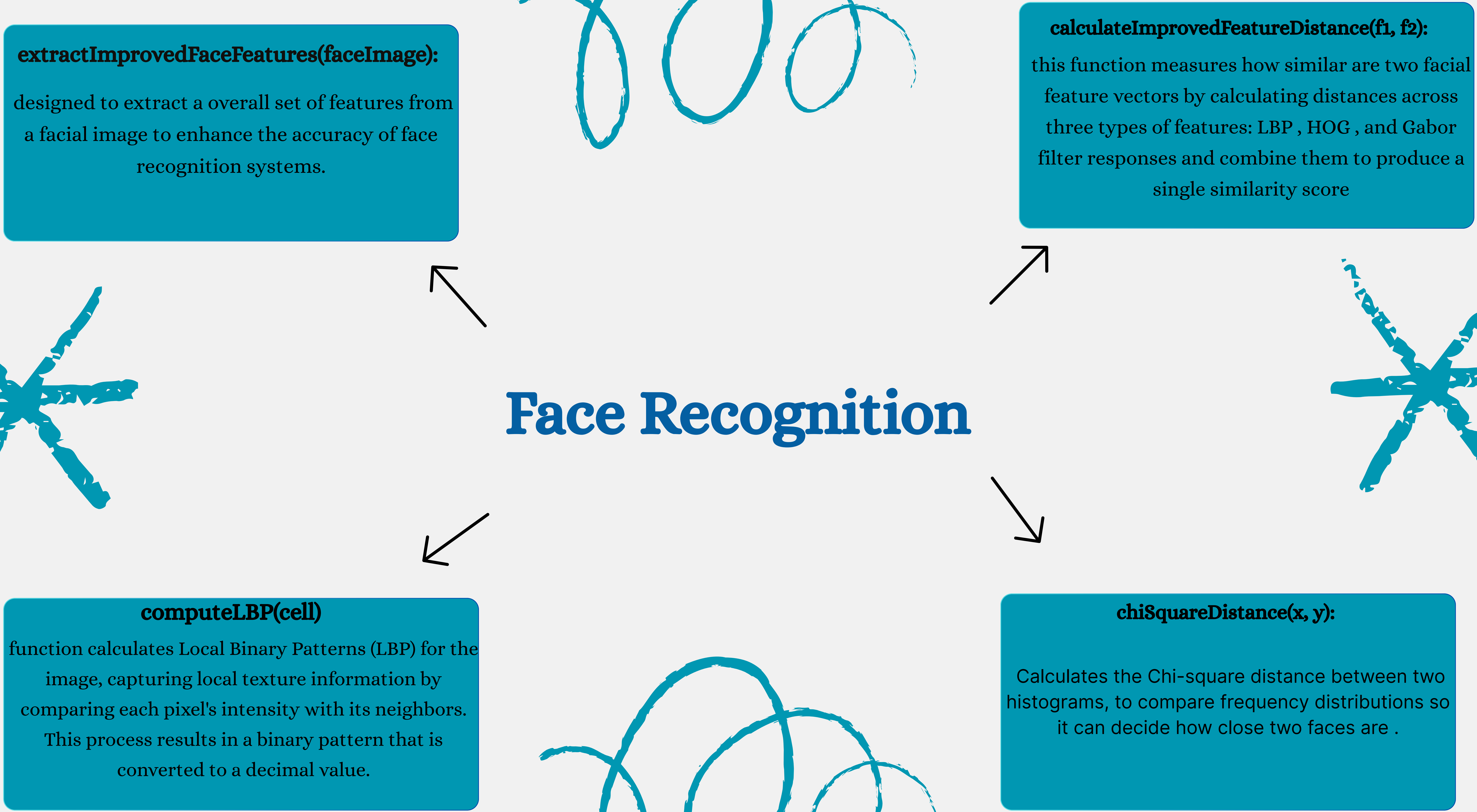
myEdgeDetection(image):
it used to detect the edges by highlighting regions with significant intensity changes

myHistEq(grayImage):
it increases the contrast to eliminate the effect of illumination

Face Detection

(rgbImage):
used to identify skin colored regions in RGB image transformations
bwlabel(cleanMask): used to identify and label connected components (objects) in a binary image

regionprops(labeledImage, 'BoundingBox', 'Area', 'Eccentricity', 'Extent'): Its used to extract features from each connected component identified by “bwlabel” function
detectEyePair(faceRegion):
is designed to verify the presence of eyes within a given facial region



Face Recognition

extractImprovedFaceFeatures(faceImage):

designed to extract a overall set of features from a facial image to enhance the accuracy of face recognition systems.

calculateImprovedFeatureDistance(f1, f2):

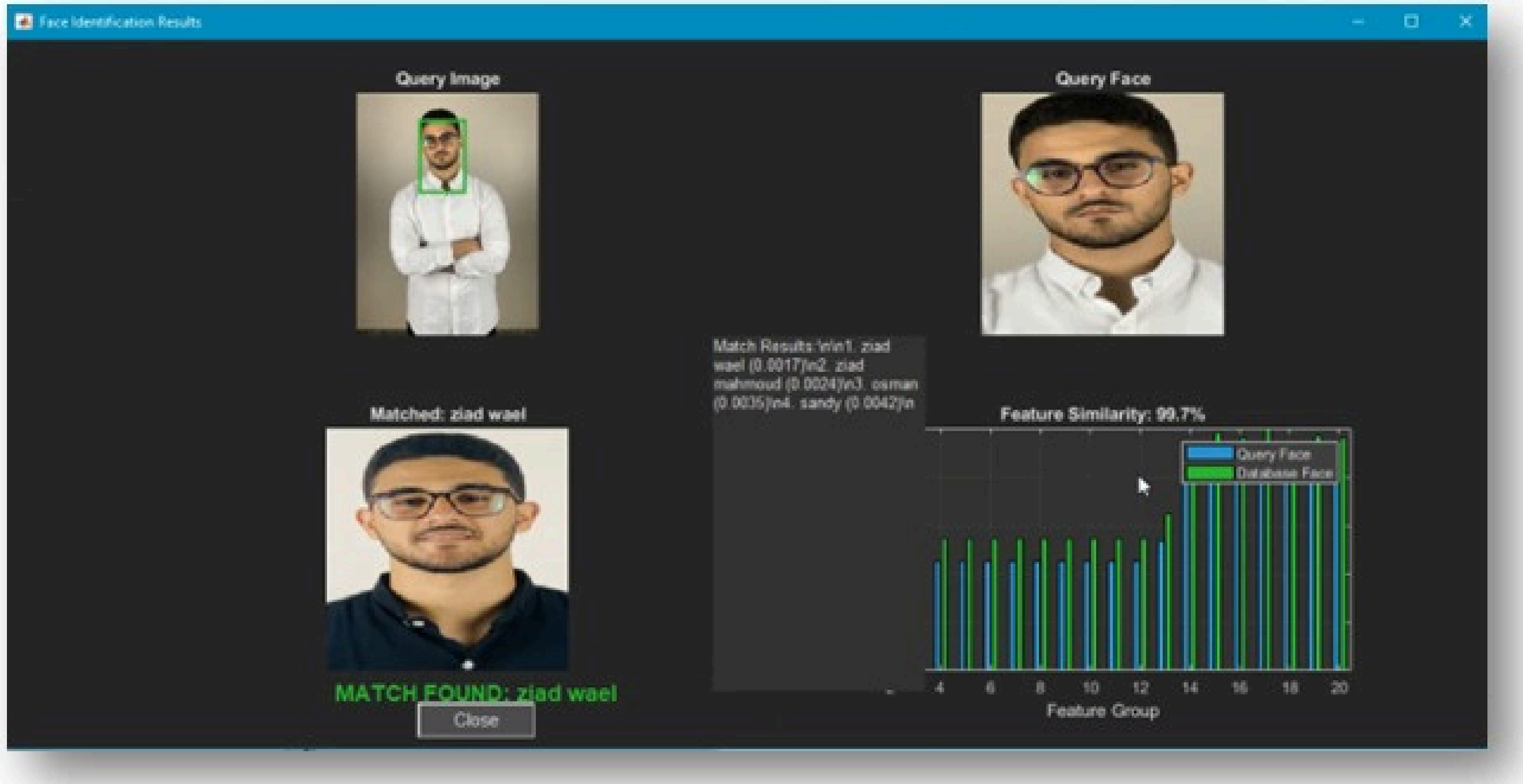
this function measures how similar are two facial feature vectors by calculating distances across three types of features: LBP , HOG , and Gabor filter responses and combine them to produce a single similarity score

computeLBP(cell)

function calculates Local Binary Patterns (LBP) for the image, capturing local texture information by comparing each pixel's intensity with its neighbors. This process results in a binary pattern that is converted to a decimal value.

chiSquareDistance(x, y):

Calculates the Chi-square distance between two histograms, to compare frequency distributions so it can decide how close two faces are .

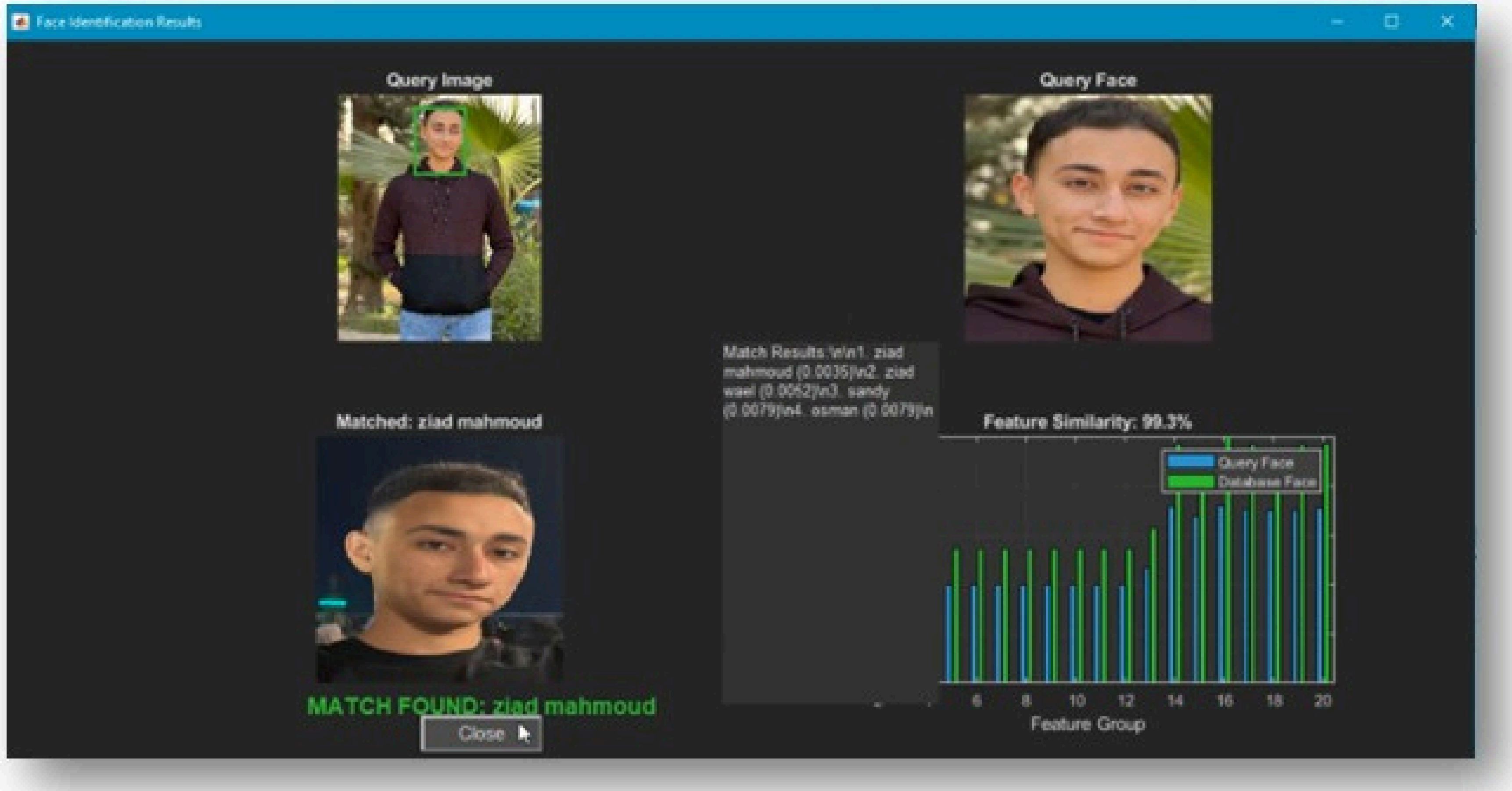


Matched: ziad wael

```
Match Results:\n\n1. ziad wael (0.0017)\n2. ziad mahmoud (0.0024)\n3. osman (0.0035)\n4. sandy (0.0042)\n
```

Feature Similarity: 99.7%

MATCH FOUND: ziad wael



Matched: ziad mahmoud

```
Match Results:\n\n1. ziad mahmoud (0.0035)\n2. ziad wael (0.0052)\n3. sandy (0.0079)\n4. osman (0.0079)\n
```

Feature Similarity: 99.3%

MATCH FOUND: ziad mahmoud

80%

Success cases

Using low-level features, the accuracy of face detection was not very high.
Many steps had to be done manually, making the code very large.
The code can recognize only one face, not multiple faces.

Manual Object Detection using MATLAB

01

Convert image to grayscale

02

Apply smoothing with a Gaussian filter

03

Detect edges using Sobel operator

04

Clean up using morphological operations

05

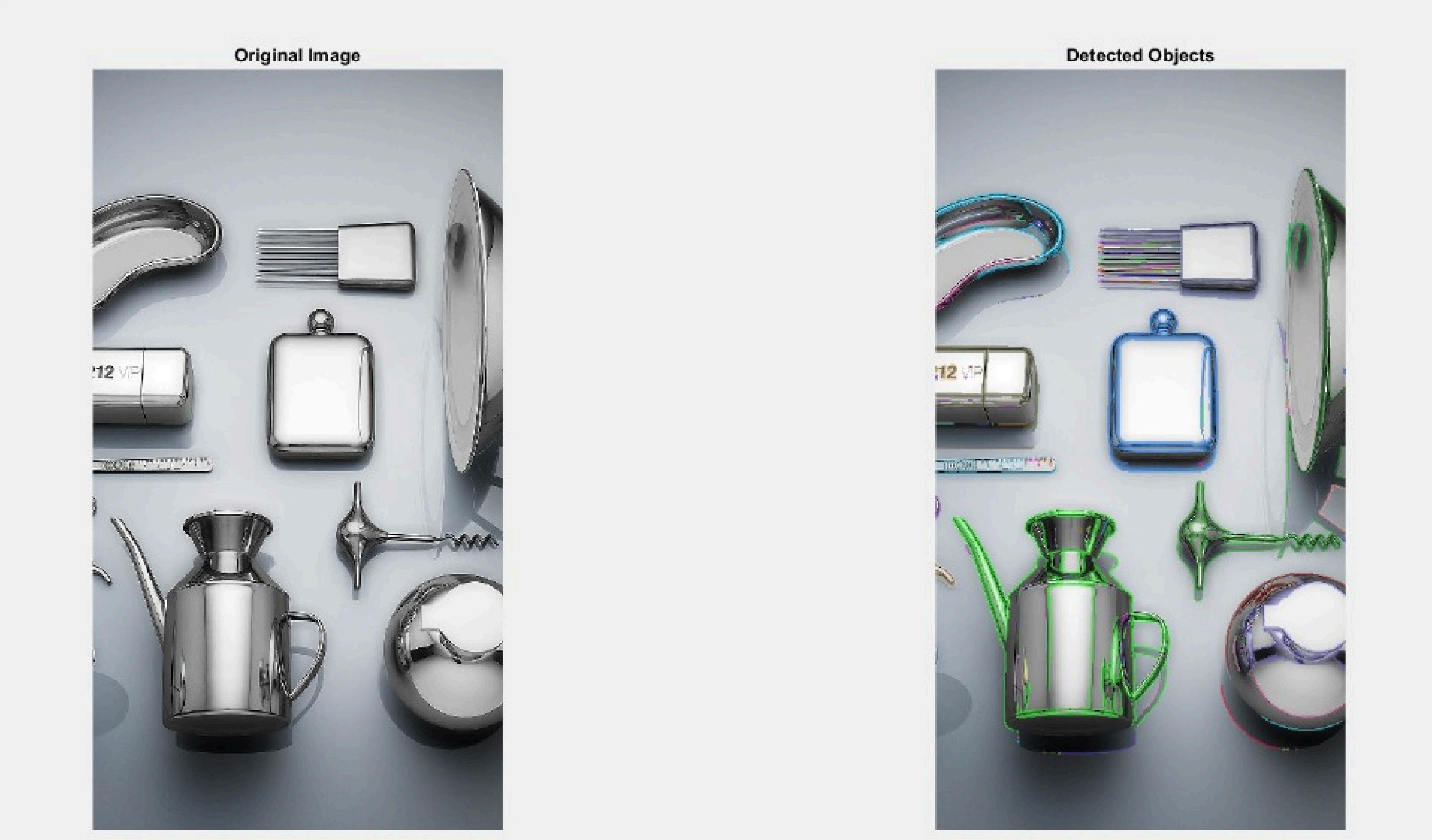
Label and count detected objects

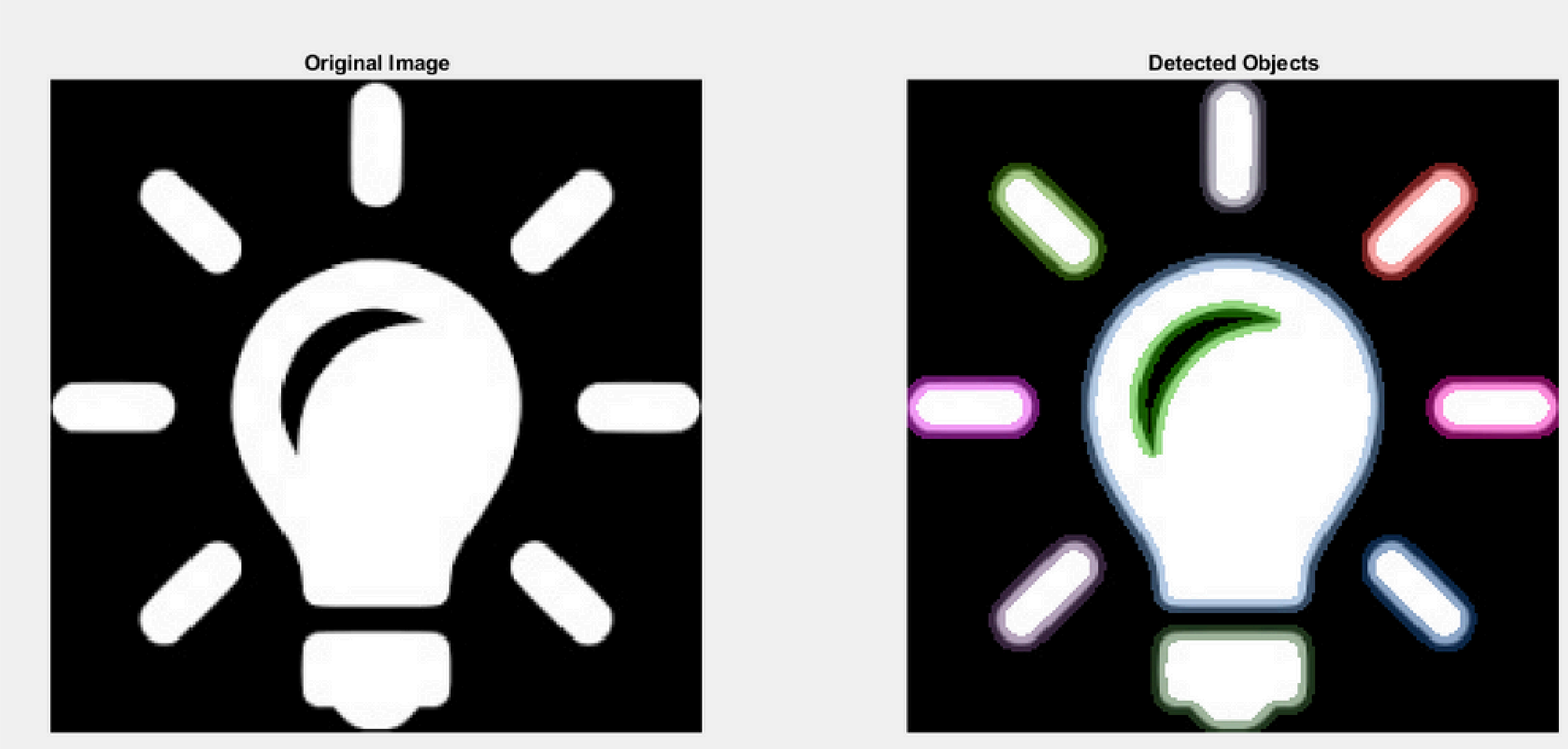
06

Visualize results

Visualizing the Result

- Overlay random colors on each detected object.
- Blend these colors with the original image for a semi-transparent effect.
- Show:
- Original image
- Image with detected objects





95%

Success cases

Objects were detected clearly.
5% built-in functions used.
Noise removed, edges detected.
Objects cleaned and colored.

Any questions?

Thank you