2ECO76

MATLAB for engineers

Term Assignment code and output

Face, eyes, Nose and lips detection using MATLAB

19BCE246-Akshat Shah

19BCE247-Devansh Shah

Code for training Data:

Face:

```
% Face Detection Steps :-
% 1.Image labelling
% 2.Training
% 3.Testing

load('faced.mat');
%  1. Load labelled data file ,which created through image labelling

Facedetect = selectLabels(gTruth,'Face');
% 2. Creating Variable facedetect in which will store labels 'face'

if isfolder(fullfile('FaceTrainingData'))
cd FaceTrainingData
else
mkdir FaceTrainingData
end
addpath('FaceTrainingData');


% 3. if else condition, if means 'if full name FaceTrainingData exist ,locate that
file ' else here if FaceTrainingData Files doesnt exits Make one and
%         it to the MATLAB Path

trainingData =
objectDetectorTrainingData(Facedetect,'SamplingFactor',1,'writeLocation','FaceTrai
ningData');

% Make variable trainingData in which will store and passing Parameters like
Facedetect that is labels , Sampling factor means Examples Face images
% if sampling factor is 2 than 2times negative images taken, Writing location = as
TrainingData Folder

Fdetector = trainACFObjectDetector(trainingData,'NumStages',20);

% detector is variable storing data of ACF Object Detector Neural network ,
Numstages= Number of Training stages,
% More stages like 10,20 takes long time to train but with higher Accuracy.
% Also For No. of Stages also Depends on Number of Positive Smaple image i.e the
image we have labelled
```

```
save('FDetector.mat','Fdetector');

% saving Detector file , so once ACF detector trained ,it can be used to detect
Faces

rmpath('FaceTrainingData');

%Saving detector file in TrainingData Folder
%Upto this 13 lines of Code , It needs to run Only Once .
%once we have save Our Neural Network 'Detector.mat' file which detects faces .
one Have saved in TrainingData folder ,
% So to use it whenever we just need to load it by specfying  its path

% Once detector is Trained.
% Above codes , Not needed to Run again and again
%
% Below Codes are to be Run.

load('FDetector.mat');

%Load Detector file , it is Pretrained Neural network for face detection
```

Eyes:

```
% Face Detection Steps :-
% 1.Image labelling
% 2.Training
% 3.Testing

load('eyesd.mat');
%  1. Load labelled data file ,which created through image labelling

Facedetect = selectLabels(gTruth,'Eyes');
% 2. Creating Variable facedetect in which will store labels 'lips'

if isfolder(fullfile('EyesTrainingData'))
cd EyesTrainingData
else
mkdir EyesTrainingData
end
addpath('EyesTrainingData');


% 3. if else condition, if means 'if full name EyesTrainingData exist ,locate that
file ' else here if EyesTrainingData Files doesnt exits Make one and
%         it to the MATLAB Path

trainingData =
objectDetectorTrainingData(Facedetect,'SamplingFactor',1,'writeLocation','EyesTrai
ningData');

% Make variable trainingData in which will store and passing Parameters like
Facedetect that is labels , Sampling factor means Examples Face images
```

```
% if sampling factor is 2 than 2times negative images taken, Writing location = as
TrainingData Folder

Edetector = trainACFObjectDetector(trainingData,'NumStages',20);

% detector is variable storing data of ACF Object Detector Neural network ,
Numstages= Number of Training stages,
% More stages like 10,20 takes long time to train but with higher Accuracy.
% Also For No. of Stages also Depends on Number of Positive Smaple image i.e the
image we have labelled

save('EDetector.mat','Edetector');

% saving Detector file , so once ACF detector trained ,it can be used to detect
Faces

rmpath('EyesTrainingData');

%Saving detector file in TrainingData Folder
%Upto this 13 lines of Code , It needs to run Only Once .
%once we have save Our Neural Network 'Detector.mat' file which detects faces .
one Have saved in TrainingData folder ,
% So to use it whenever we just need to load it by specfying  its path

% Once detector is Trained.
% Above codes , Not needed to Run again and again
%
% Below Codes are to be Run.

load('EDetector.mat');

%Load Detector file , it is Pretrained Neural network for face detection
```

Nose:

```
% Face Detection Steps :-
% 1.Image labelling
% 2.Training
% 3.Testing

load('nosed.mat');
%  1. Load labelled data file ,which created through image labelling

Facedetect = selectLabels(gTruth,'Nose');
% 2. Creating Variable facedetect in which will store labels 'lips'

if isfolder(fullfile('NoseTrainingData'))
cd NoseTrainingData
else
mkdir NoseTrainingData
end
addpath('NoseTrainingData');
```

```
% 3. if else condition, if means 'if full name NoseTrainingData exist ,locate that
file ' else here if NoseTrainingData Files doesnt exits Make one and
%         it to the MATLAB Path

trainingData =
objectDetectorTrainingData(Facedetect,'SamplingFactor',1,'writeLocation','NoseTrai
ningData');

% Make variable trainingData in which will store and passing Parameters like
Facedetect that is labels , Sampling factor means Examples Face images
% if sampling factor is 2 than 2times negative images taken, Writing location = as
TrainingData Folder

Ndetector = trainACFObjectDetector(trainingData,'NumStages',20);

% detector is variable storing data of ACF Object Detector Neural network ,
Numstages= Number of Training stages,
% More stages like 10,20 takes long time to train but with higher Accuracy.
% Also For No. of Stages also Depends on Number of Positive Smaple image i.e the
image we have labelled

save('NDetector.mat','Ndetector');

% saving Detector file , so once ACF detector trained ,it can be used to detect
Faces

rmpath('NoseTrainingData');

%Saving detector file in TrainingData Folder
%Upto this 13 lines of Code , It needs to run Only Once .
%once we have save Our Neural Network 'Detector.mat' file which detects faces .
one Have saved in TrainingData folder ,
% So to use it whenever we just need to load it by specfying  its path

% Once detector is Trained.
% Above codes , Not needed to Run again and again
%
% Below Codes are to be Run.

load('NDetector.mat');

%Load Detector file , it is Pretrained Neural network for face detection
```

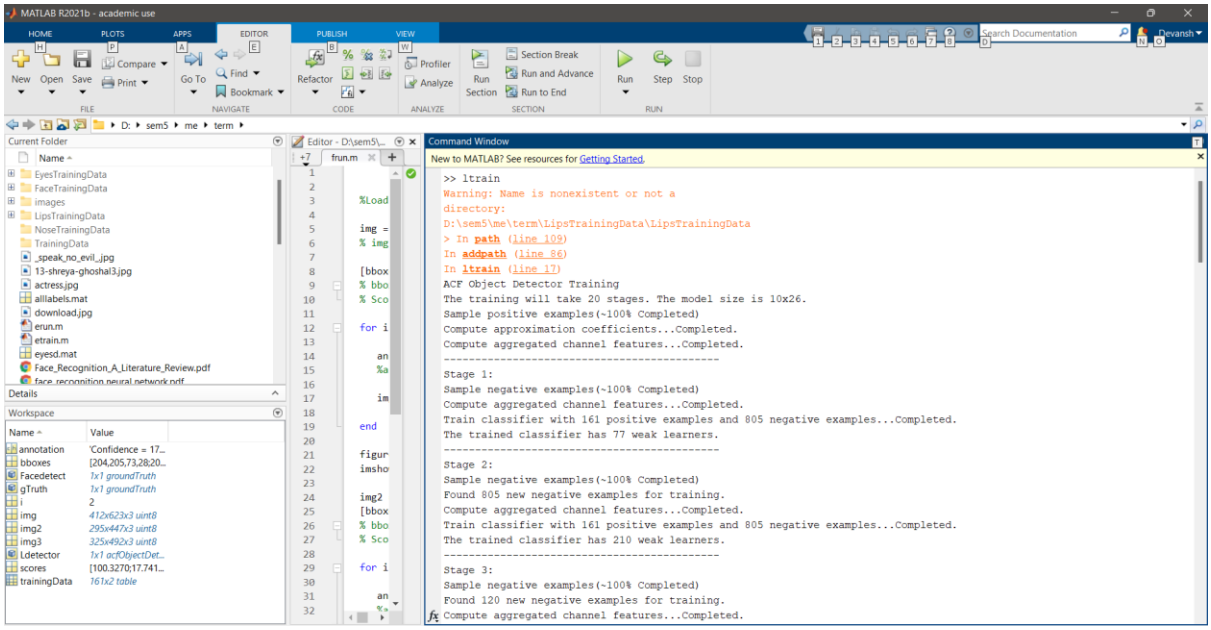
Lips:

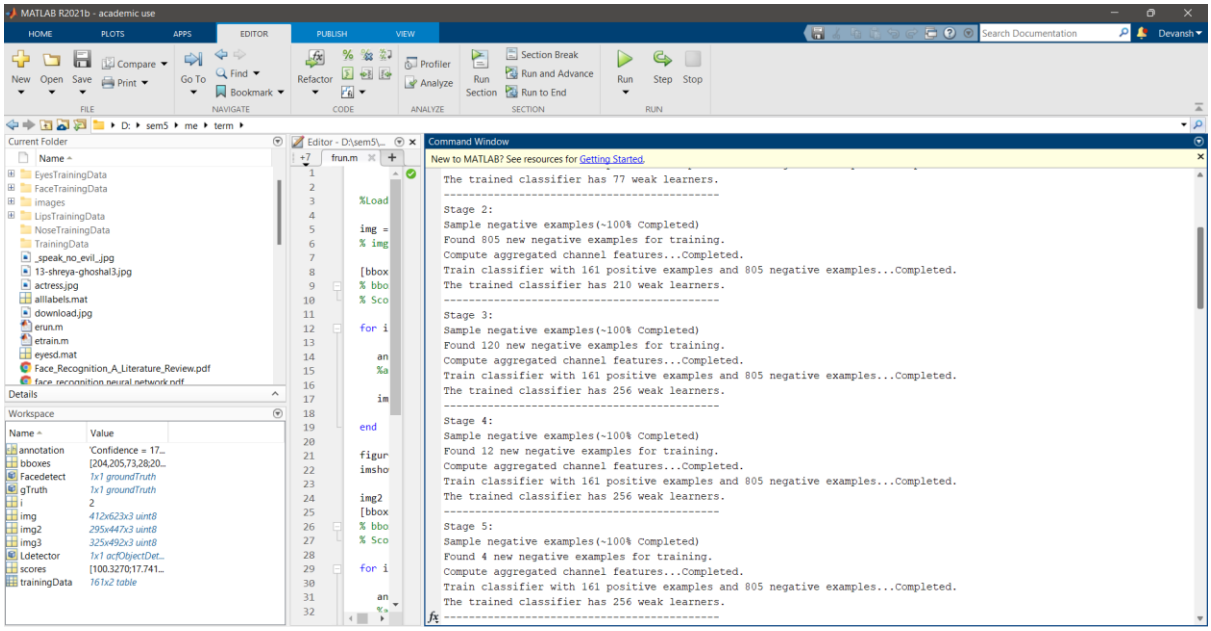
```
% Face Detection Steps :-
% 1.Image labelling
% 2.Training
% 3.Testing

load('lipd.mat');
%  1. Load labelled data file ,which created through image labelling

Facedetect = selectLabels(gTruth,'Lips');
```

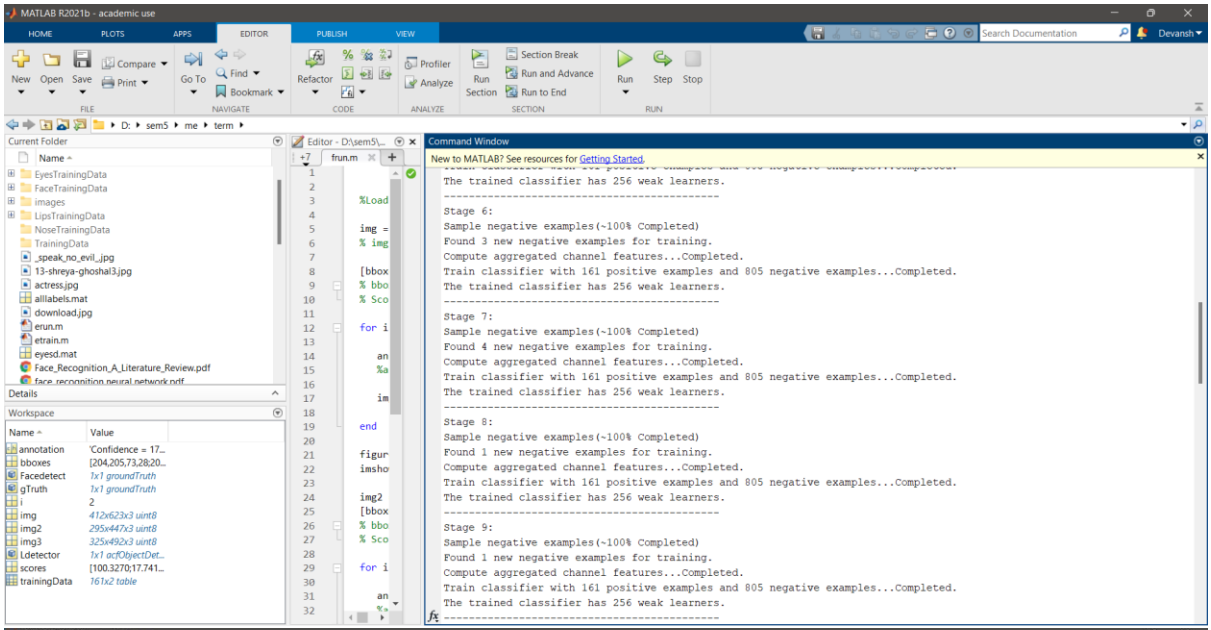
```
% 2. Creating Variable facedetect in which will store labels 'lips'

if isfolder(fullfile('LipsTrainingData'))
cd LipsTrainingData
else
mkdir LipsTrainingData
end
addpath('LipsTrainingData');


% 3. if else condition, if means 'if full name LipsTrainingData exist ,locate that
file ' else here if LipsTrainingData Files doesnt exits Make one and
%         it to the MATLAB Path

trainingData =
objectDetectorTrainingData(Facedetect,'SamplingFactor',1,'writeLocation','LipsTrai
ningData');

% Make variable trainingData in which will store and passing Parameters like
Facedetect that is labels , Sampling factor means Examples Face images
% if sampling factor is 2 than 2times negative images taken, Writing location = as
TrainingData Folder

Ldetector = trainACFObjectDetector(trainingData,'NumStages',20);

% detector is variable storing data of ACF Object Detector Neural network ,
Numstages= Number of Training stages,
% More stages like 10,20 takes long time to train but with higher Accuracy.
% Also For No. of Stages also Depends on Number of Positive Smaple image i.e the
image we have labelled

save('LDetector.mat','Ldetector');

% saving Detector file , so once ACF detector trained ,it can be used to detect
Faces

rmpath('LipsTrainingData');

%Saving detector file in TrainingData Folder
%Upto this 13 lines of Code , It needs to run Only Once .
%once we have save Our Neural Network 'Detector.mat' file which detects faces .
one Have saved in TrainingData folder ,
% So to use it whenever we just need to load it by specfying  its path

% Once detector is Trained.
% Above codes , Not needed to Run again and again
%
% Below Codes are to be Run.

load('LDetector.mat');

%Load Detector file , it is Pretrained Neural network for face detection
```

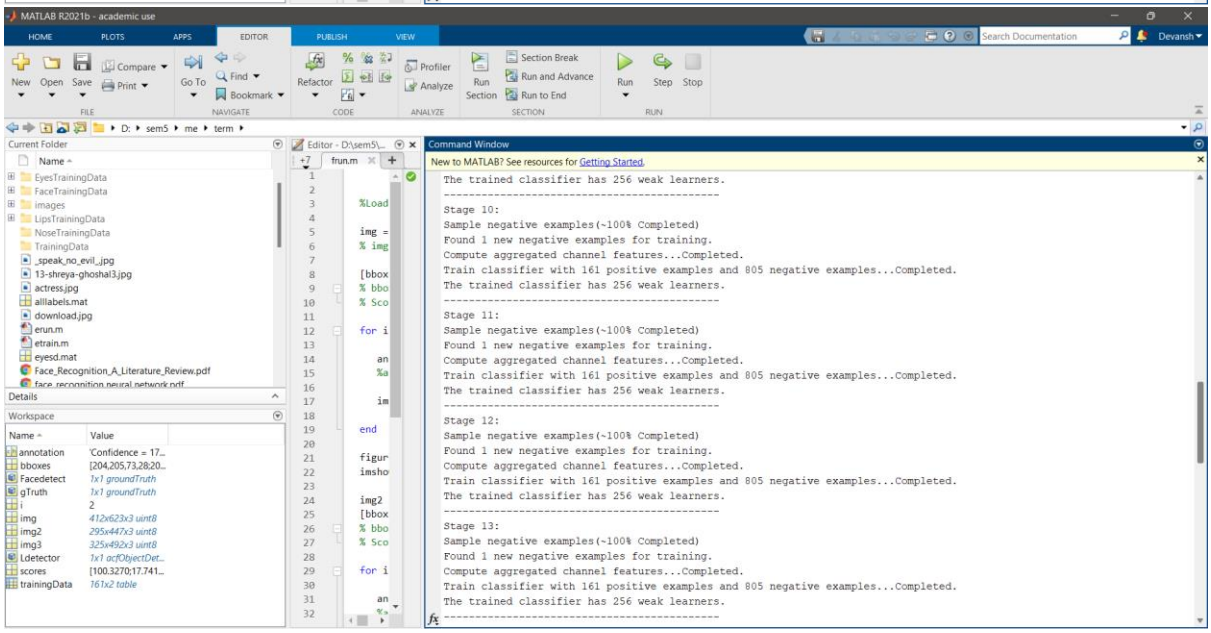
Output for training phase(it's the same for training each data):

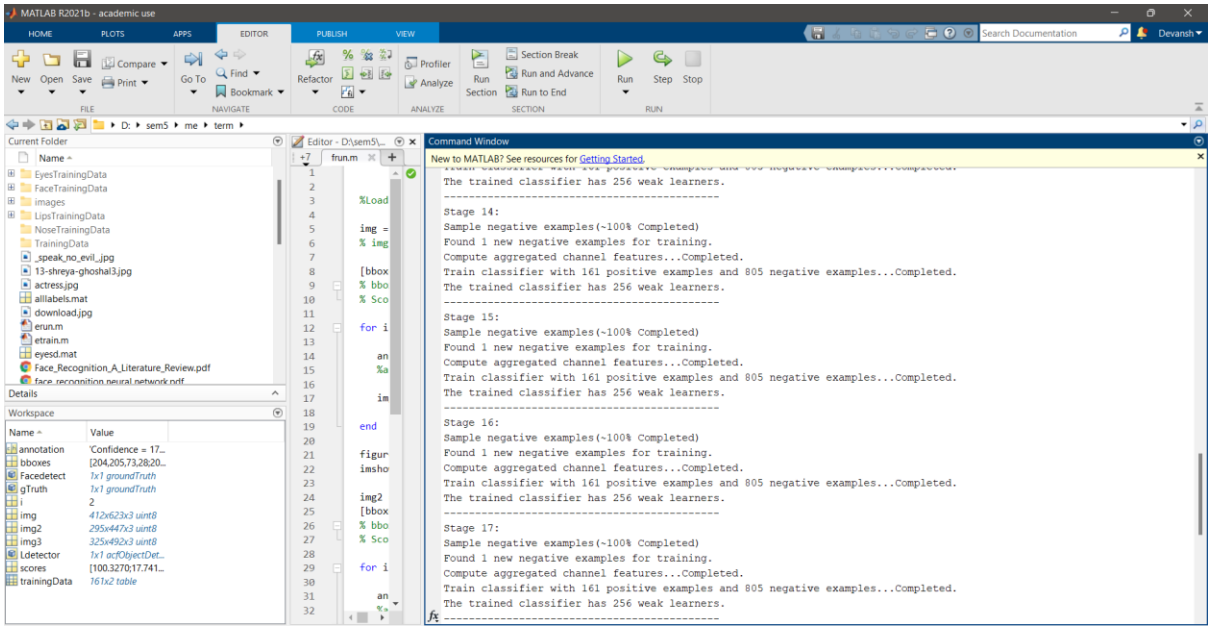
- 20 different stages are ran for better accuracy.

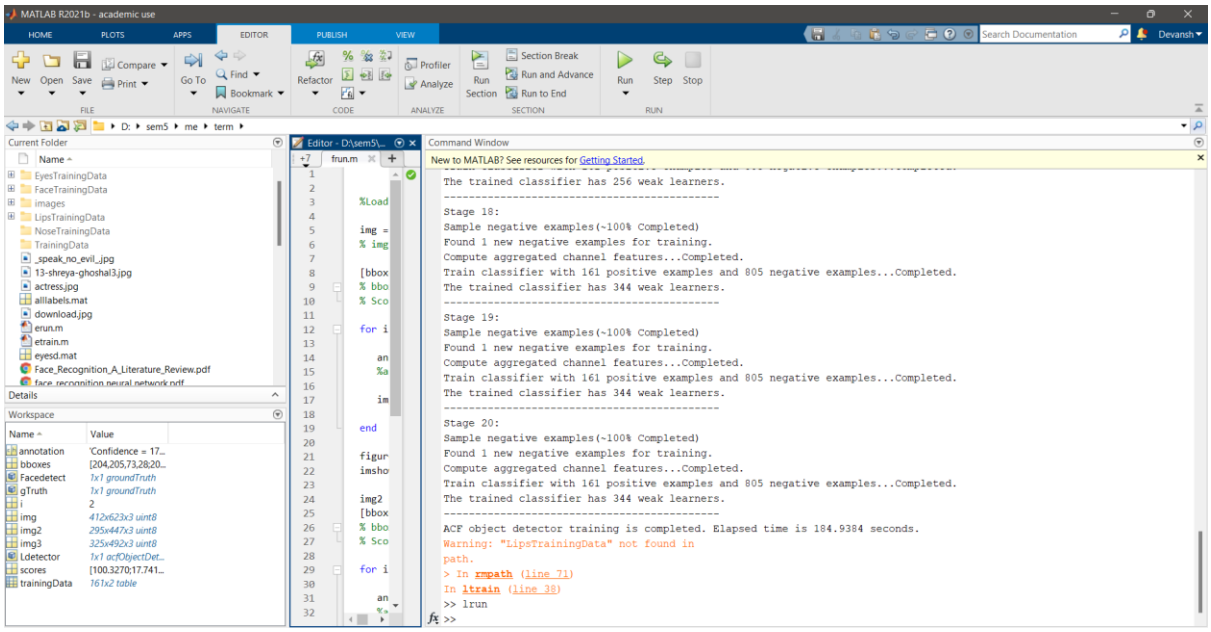





MATLAB R2021b - academic use
HOME
PLOTS
APPS
EDITOR
PUBLISH
VIEW
Search Documentation
Devansh
New
Open
Save
Compare
Print
Go To
Find
Bookmark
Refactor
Profiler
Analyze
Run Section
Section Break
Run and Advance
Run to End
Run
Step
Stop
FILE
NAVIGATE
CODE
ANALYZE
SECTION
RUN
D: ▸ sem5 ▸ me ▸ term ▸
Current Folder
Name
EyesTrainingData
FaceTrainingData
images
LipsTrainingData
NoseTrainingData
TrainingData
_speak_no_evil_.jpg
13-shreya-ghoshal3.jpg
actress.jpg
alllabels.mat
download.jpg
erun.m
etrain.m
eyesd.mat
Face_Recognition_A_Literature_Review.pdf
face_recognition_neural_network.pdf
Details
Workspace
Name
Value
annotation
'Confidence = 17...
bboxes
[204,205,73,28;20...
Facedetect
1x1 groundTruth
gTruth
1x1 groundTruth
i
2
img
412x623x3 uint8
img2
295x447x3 uint8
img3
325x492x3 uint8
Ldetector
1x1 acfObjectDet...
scores
[100.3270;17.741...
trainingData
161x2 table
Editor - D:\sem5\...
frun.m
%Load
img =
% img
[bbox
% bbo
% Sco
for i
an
%a
im
end
figur
imsho
img2
[bbox
% bbo
% Sco
for i
an
Command Window
New to MATLAB? See resources for Getting Started.
The trained classifier has 256 weak learners.
Stage 6:
Sample negative examples(~100% Completed)
Found 3 new negative examples for training.
Compute aggregated channel features...Completed.
Train classifier with 161 positive examples and 805 negative examples...Completed.
The trained classifier has 256 weak learners.
Stage 7:
Sample negative examples(~100% Completed)
Found 4 new negative examples for training.
Compute aggregated channel features...Completed.
Train classifier with 161 positive examples and 805 negative examples...Completed.
The trained classifier has 256 weak learners.
Stage 8:
Sample negative examples(~100% Completed)
Found 1 new negative examples for training.
Compute aggregated channel features...Completed.
Train classifier with 161 positive examples and 805 negative examples...Completed.
The trained classifier has 256 weak learners.
Stage 9:
Sample negative examples(~100% Completed)
Found 1 new negative examples for training.
Compute aggregated channel features...Completed.
Train classifier with 161 positive examples and 805 negative examples...Completed.
The trained classifier has 256 weak learners.


MATLAB R2021b - academic use
HOME
PLOTS
APPS
EDITOR
PUBLISH
VIEW
Search Documentation
Devansh
New
Open
Save
Compare
Print
Go To
Find
Bookmark
Refactor
Profiler
Analyze
Run Section
Section Break
Run and Advance
Run to End
Run
Step
Stop
FILE
NAVIGATE
CODE
ANALYZE
SECTION
RUN
D: ▸ sem5 ▸ me ▸ term ▸
Current Folder
Name
EyesTrainingData
FaceTrainingData
images
LipsTrainingData
NoseTrainingData
TrainingData
_speak_no_evil_.jpg
13-shreya-ghoshal3.jpg
actress.jpg
alllabels.mat
download.jpg
erun.m
etrain.m
eyesd.mat
Face_Recognition_A_Literature_Review.pdf
face_recognition_neural_network.pdf
Details
Workspace
Name
Value
annotation
'Confidence = 17...
bboxes
[204,205,73,28;20...
Facedetect
1x1 groundTruth
gTruth
1x1 groundTruth
i
2
img
412x623x3 uint8
img2
295x447x3 uint8
img3
325x492x3 uint8
Ldetector
1x1 acfObjectDet...
scores
[100.3270;17.741...
trainingData
161x2 table
Editor - D:\sem5\...
frun.m
%Load
img =
% img
[bbox
% bbo
% Sco
for i
an
%a
im
end
figur
imsho
img2
[bbox
% bbo
% Sco
for i
an
Command Window
New to MATLAB? See resources for Getting Started.
The trained classifier has 256 weak learners.
Stage 10:
Sample negative examples(~100% Completed)
Found 1 new negative examples for training.
Compute aggregated channel features...Completed.
Train classifier with 161 positive examples and 805 negative examples...Completed.
The trained classifier has 256 weak learners.
Stage 11:
Sample negative examples(~100% Completed)
Found 1 new negative examples for training.
Compute aggregated channel features...Completed.
Train classifier with 161 positive examples and 805 negative examples...Completed.
The trained classifier has 256 weak learners.
Stage 12:
Sample negative examples(~100% Completed)
Found 1 new negative examples for training.
Compute aggregated channel features...Completed.
Train classifier with 161 positive examples and 805 negative examples...Completed.
The trained classifier has 256 weak learners.
Stage 13:
Sample negative examples(~100% Completed)
Found 1 new negative examples for training.
Compute aggregated channel features...Completed.
Train classifier with 161 positive examples and 805 negative examples...Completed.
The trained classifier has 256 weak learners.


MATLAB R2021b - academic use
HOME
PLOTS
APPS
EDITOR
PUBLISH
VIEW
Search Documentation
Devansh
New
Open
Save
Compare
Print
Go To
Find
Bookmark
Refactor
Profiler
Analyze
Run Section
Section Break
Run and Advance
Run to End
Run
Step
Stop
FILE
NAVIGATE
CODE
ANALYZE
SECTION
RUN
D: ▸ sem5 ▸ me ▸ term ▸
Current Folder
Name
EyesTrainingData
FaceTrainingData
images
LipsTrainingData
NoseTrainingData
TrainingData
_speak_no_evil_.jpg
13-shreya-ghoshal3.jpg
actress.jpg
alllabels.mat
download.jpg
erun.m
etrain.m
eyesd.mat
Face_Recognition_A_Literature_Review.pdf
face_recognition_neural_network.pdf
Details
Workspace
Name
Value
annotation
'Confidence = 17...
bboxes
[204,205,73,28;20...
Facedetect
1x1 groundTruth
gTruth
1x1 groundTruth
i
2
img
412x623x3 uint8
img2
295x447x3 uint8
img3
325x492x3 uint8
Ldetector
1x1 acfObjectDet...
scores
[100.3270;17.741...
trainingData
161x2 table
Editor - D:\sem5\...
frun.m
%Load
img =
% img
[bbox
% bbo
% Sco
for i
an
%a
im
end
figur
imsho
img2
[bbox
% bbo
% Sco
for i
an
Command Window
New to MATLAB? See resources for Getting Started.
The trained classifier has 256 weak learners.
Stage 14:
Sample negative examples(~100% Completed)
Found 1 new negative examples for training.
Compute aggregated channel features...Completed.
Train classifier with 161 positive examples and 805 negative examples...Completed.
The trained classifier has 256 weak learners.
Stage 15:
Sample negative examples(~100% Completed)
Found 1 new negative examples for training.
Compute aggregated channel features...Completed.
Train classifier with 161 positive examples and 805 negative examples...Completed.
The trained classifier has 256 weak learners.
Stage 16:
Sample negative examples(~100% Completed)
Found 1 new negative examples for training.
Compute aggregated channel features...Completed.
Train classifier with 161 positive examples and 805 negative examples...Completed.
The trained classifier has 256 weak learners.
Stage 17:
Sample negative examples(~100% Completed)
Found 1 new negative examples for training.
Compute aggregated channel features...Completed.
Train classifier with 161 positive examples and 805 negative examples...Completed.
The trained classifier has 256 weak learners.


MATLAB R2021b - academic use
HOME
PLOTS
APPS
EDITOR
PUBLISH
VIEW
Search Documentation
Devansh
New
Open
Save
Compare
Print
Go To
Find
Bookmark
Refactor
Profiler
Analyze
Run Section
Section Break
Run and Advance
Run to End
Run
Step
Stop
FILE
NAVIGATE
CODE
ANALYZE
SECTION
RUN
D: ▸ sem5 ▸ me ▸ term ▸
Current Folder
Name
EyesTrainingData
FaceTrainingData
images
LipsTrainingData
NoseTrainingData
TrainingData
_speak_no_evil_.jpg
13-shreya-ghoshal3.jpg
actress.jpg
alllabels.mat
download.jpg
erun.m
etrain.m
eyesd.mat
Face_Recognition_A_Literature_Review.pdf
face_recognition_neural_network.pdf
Details
Workspace
Name
Value
annotation
'Confidence = 17...
bboxes
[204,205,73,28;20...
Facedetect
1x1 groundTruth
gTruth
1x1 groundTruth
i
2
img
412x623x3 uint8
img2
295x447x3 uint8
img3
325x492x3 uint8
Ldetector
1x1 acfObjectDet...
scores
[100.3270;17.741...
trainingData
161x2 table
Editor - D:\sem5\...
frun.m
%Load
img =
% img
[bbox
% bbo
% Sco
for i
an
%a
im
end
figur
imsho
img2
[bbox
% bbo
% Sco
for i
an
Command Window
New to MATLAB? See resources for Getting Started.
The trained classifier has 256 weak learners.
Stage 18:
Sample negative examples(~100% Completed)
Found 1 new negative examples for training.
Compute aggregated channel features...Completed.
Train classifier with 161 positive examples and 805 negative examples...Completed.
The trained classifier has 344 weak learners.
Stage 19:
Sample negative examples(~100% Completed)
Found 1 new negative examples for training.
Compute aggregated channel features...Completed.
Train classifier with 161 positive examples and 805 negative examples...Completed.
The trained classifier has 344 weak learners.
Stage 20:
Sample negative examples(~100% Completed)
Found 1 new negative examples for training.
Compute aggregated channel features...Completed.
Train classifier with 161 positive examples and 805 negative examples...Completed.
The trained classifier has 344 weak learners.
ACF object detector training is completed. Elapsed time is 184.9384 seconds.
Warning: "LipsTrainingData" not found in
path.
> In rmpath (line 71)
In ltrain (line 38)
>> lrun
>>

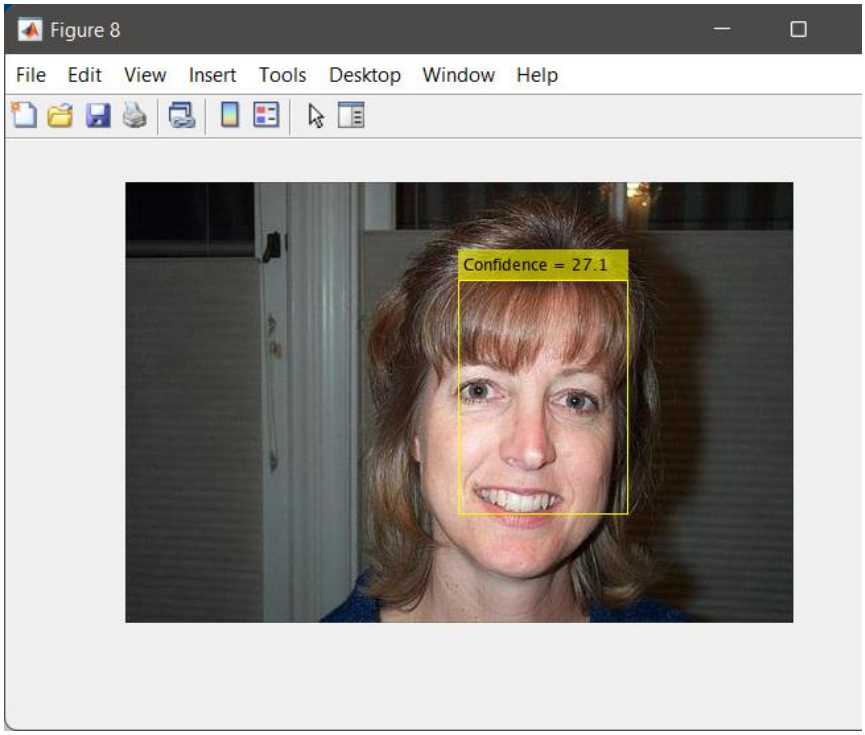
Code for Testing Phase:

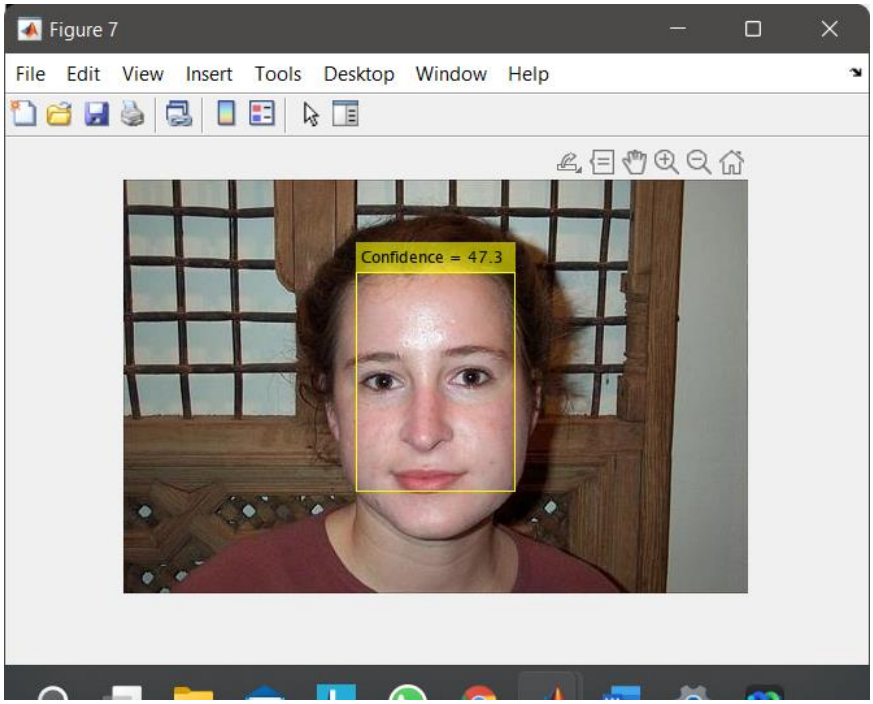
Face:

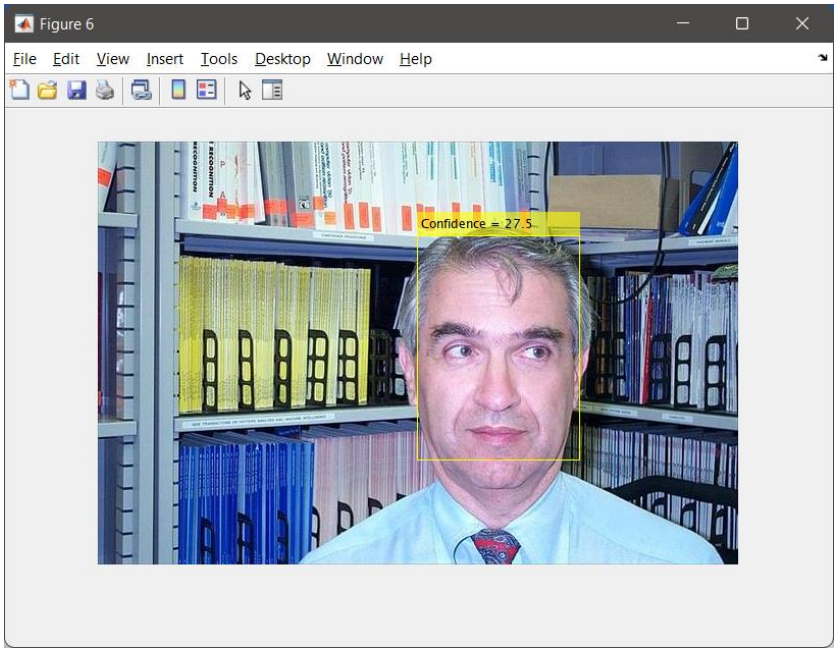
```
%Load Detector file , it is Pretrained Neural network for face detection

img = imread('images\Fd\image_0102.jpg');
% img is Variable , imread is function for reading Image.

[bboxes,scores] = detect(Fdetector,img);
% bboxes = Bounding Boxes which surrounds Face -Rectangle Box
% Scores = Confidence that is how sure a Detector is for identifying Human Face

for i = 1:length(scores)

   annotation = sprintf('Confidence = %.1f',scores(i));
   %annotation is labels like face and Confidence in percentage

   img = insertObjectAnnotation(img,'rectangle',bboxes(i,:),annotation);

end

figure
imshow(img);

img2 = imread('D:\sem5\me\term\images\Fd\image_0120.jpg');
[bboxes,scores] = detect(Fdetector,img2);
% bboxes = Bounding Boxes which surrounds Face -Rectangle Box
% Scores = Confidence that is how sure a Detector is for identifying Human Face

for i = 1:length(scores)

   annotation = sprintf('Confidence = %.1f',scores(i));
   %annotation is labels like face and Confidence in percentage

   img2 = insertObjectAnnotation(img2,'rectangle',bboxes(i,:),annotation);

end

figure
imshow(img2);

img3 = imread('D:\sem5\me\term\images\Fd\image_0175.jpg');
[bboxes,scores] = detect(Fdetector,img3);
% bboxes = Bounding Boxes which surrounds Face -Rectangle Box
% Scores = Confidence that is how sure a Detector is for identifying Human Face

for i = 1:length(scores)

   annotation = sprintf('Confidence = %.1f',scores(i));
   %annotation is labels like face and Confidence in percentage

   img3 = insertObjectAnnotation(img3,'rectangle',bboxes(i,:),annotation);

end

figure
```

```
imshow(img3);
```

Output for face:





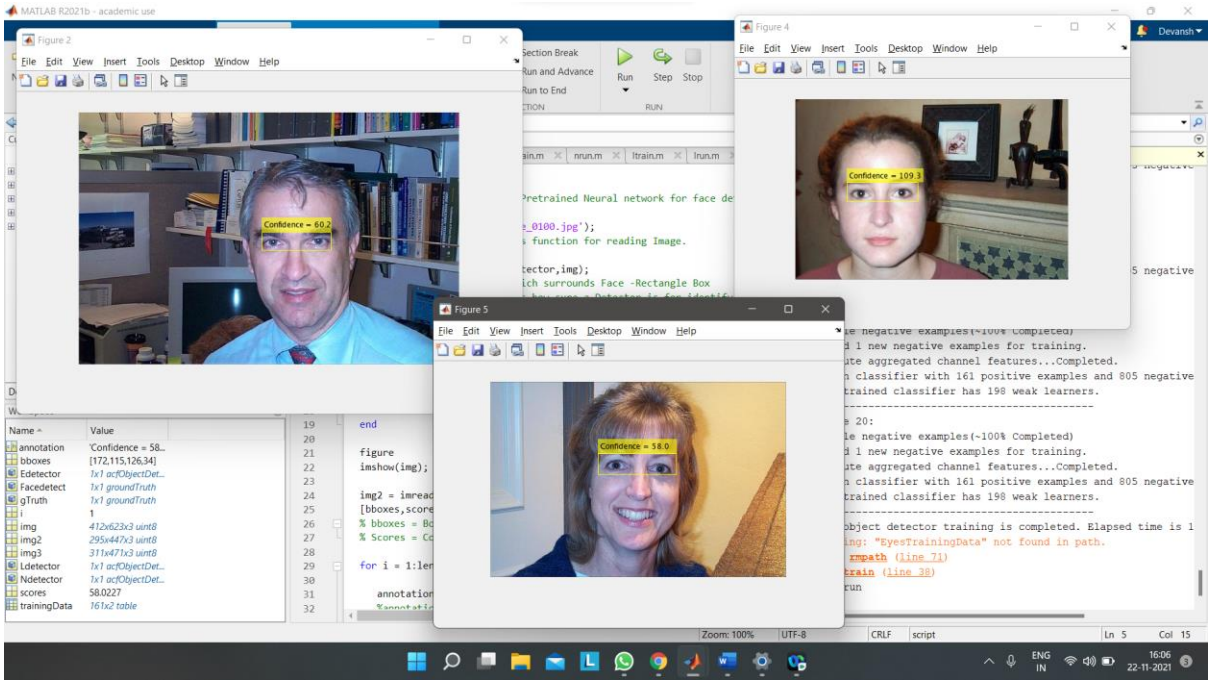


Eyes:

```
%Load Detector file , it is Pretrained Neural network for face detection

img = imread('images\Fd\image_0100.jpg');
% img is Variable , imread is function for reading Image.

[bboxes,scores] = detect(Edetector,img);
% bboxes = Bounding Boxes which surrounds Face -Rectangle Box
% Scores = Confidence that is how sure a Detector is for identifying Human Face

for i = 1:length(scores)

   annotation = sprintf('Confidence = %.1f',scores(i));
   %annotation is labels like face and Confidence in percentage

   img = insertObjectAnnotation(img,'rectangle',bboxes(i,:),annotation);

end

figure
imshow(img);

img2 = imread('D:\sem5\me\term\images\Fd\image_0115.jpg');
[bboxes,scores] = detect(Edetector,img2);
% bboxes = Bounding Boxes which surrounds Face -Rectangle Box
% Scores = Confidence that is how sure a Detector is for identifying Human Face

for i = 1:length(scores)

   annotation = sprintf('Confidence = %.1f',scores(i));
   %annotation is labels like face and Confidence in percentage

   img2 = insertObjectAnnotation(img2,'rectangle',bboxes(i,:),annotation);

end
```

```
figure
imshow(img2);

img3 = imread('D:\sem5\me\term\images\Fd\image_0180.jpg');
[bboxes,scores] = detect(Edetector,img3);
% bboxes = Bounding Boxes which surrounds Face -Rectangle Box
% Scores = Confidence that is how sure a Detector is for identifying Human Face

for i = 1:length(scores)

   annotation = sprintf('Confidence = %.1f',scores(i));
   %annotation is labels like face and Confidence in percentage

   img3 = insertObjectAnnotation(img3,'rectangle',bboxes(i,:),annotation);

end

figure
imshow(img3);
```

Output for Eyes:

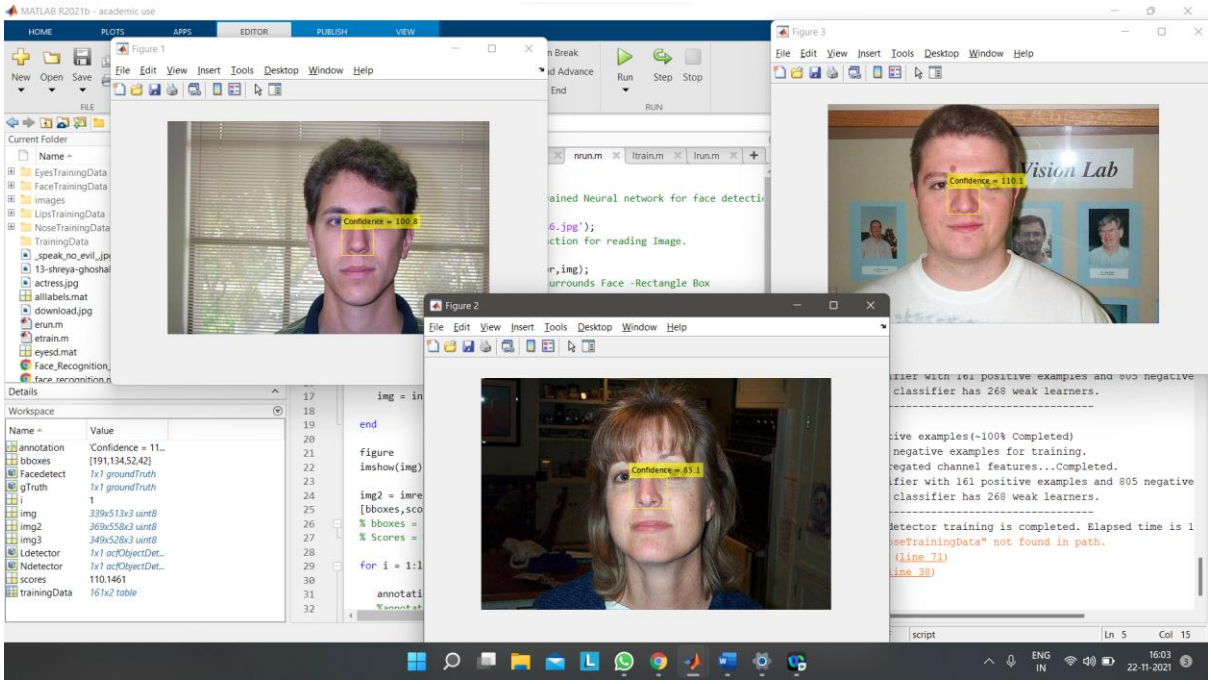


Nose:

```
%Load Detector file , it is Pretrained Neural network for face detection

img = imread('images\Fd\image_0046.jpg');
% img is Variable , imread is function for reading Image.

[bboxes,scores] = detect(Ndetector,img);
% bboxes = Bounding Boxes which surrounds Face -Rectangle Box
```

```
% Scores = Confidence that is how sure a Detector is for identifying Human Face

for i = 1:length(scores)

   annotation = sprintf('Confidence = %.1f',scores(i));
   %annotation is labels like face and Confidence in percentage

   img = insertObjectAnnotation(img,'rectangle',bboxes(i,:),annotation);

end

figure
imshow(img);

img2 = imread('D:\sem5\me\term\images\Fd\image_0190.jpg');
[bboxes,scores] = detect(Ndetector,img2);
% bboxes = Bounding Boxes which surrounds Face -Rectangle Box
% Scores = Confidence that is how sure a Detector is for identifying Human Face

for i = 1:length(scores)

   annotation = sprintf('Confidence = %.1f',scores(i));
   %annotation is labels like face and Confidence in percentage

   img2 = insertObjectAnnotation(img2,'rectangle',bboxes(i,:),annotation);

end

figure
imshow(img2);

img3 = imread('D:\sem5\me\term\images\Fd\image_0048.jpg');
[bboxes,scores] = detect(Ndetector,img3);
% bboxes = Bounding Boxes which surrounds Face -Rectangle Box
% Scores = Confidence that is how sure a Detector is for identifying Human Face

for i = 1:length(scores)

   annotation = sprintf('Confidence = %.1f',scores(i));
   %annotation is labels like face and Confidence in percentage

   img3 = insertObjectAnnotation(img3,'rectangle',bboxes(i,:),annotation);

end

figure
imshow(img3);
```

Output for nose:



Lips:

```
%Load Detector file , it is Pretrained Neural network for face detection

img = imread('images\Fd\image_0100.jpg');
% img is Variable , imread is function for reading Image.

[bboxes,scores] = detect(Ldetector,img);
% bboxes = Bounding Boxes which surrounds Face -Rectangle Box
% Scores = Confidence that is how sure a Detector is for identifying Human Face

for i = 1:length(scores)

   annotation = sprintf('Confidence = %.1f',scores(i));
   %annotation is labels like face and Confidence in percentage

   img = insertObjectAnnotation(img,'rectangle',bboxes(i,:),annotation);

end

figure
imshow(img);

img2 = imread('D:\sem5\me\term\images\Fd\image_0115.jpg');
[bboxes,scores] = detect(Ldetector,img2);
% bboxes = Bounding Boxes which surrounds Face -Rectangle Box
% Scores = Confidence that is how sure a Detector is for identifying Human Face

for i = 1:length(scores)

   annotation = sprintf('Confidence = %.1f',scores(i));
   %annotation is labels like face and Confidence in percentage

   img2 = insertObjectAnnotation(img2,'rectangle',bboxes(i,:),annotation);
```

```
end

figure
imshow(img2);

img3 = imread('D:\sem5\me\term\images\Fd\image_0140.jpg');
[bboxes,scores] = detect(Ldetector,img3);
% bboxes = Bounding Boxes which surrounds Face -Rectangle Box
% Scores = Confidence that is how sure a Detector is for identifying Human Face

for i = 1:length(scores)

   annotation = sprintf('Confidence = %.1f',scores(i));
   %annotation is labels like face and Confidence in percentage

   img3 = insertObjectAnnotation(img3,'rectangle',bboxes(i,:),annotation);

end

figure
imshow(img3);
```

Output for Lips: