Computer vision practical assignment 4

Face Detection using Viola-Jones Approach

20321233 zou letian

20321308 cao Xinyang

20321309 chen hanzheng

1.Objective

Study of Viola-Jones approach for detection of faces and part of bodies in the images.

2.Procedure of Practical Assignment Performing

1. *Faces detection.* Select three arbitrary images contains several faces. Try to use images with a different number of faces and different scales. Perform to search faces using Viola-Jones approach. Calculate the number of found faces on each image.

2. *Body parts detection.* Select three arbitrary images contains several faces. Try to use images with a different number of faces and different scales. Perform to search at least two parts of bodies in the one image (e.g. eyes, mouths, noses). To increase the accuracy use ROI (upper part of bodies or faces). Calculate the found elements in each category.

3. *Optional 1.* Implement the face detection in videostream using pre-recorded video with faces.

4. *Optional 2.* Implement the face detection in live videostream using web-camera.

Note. Please note that when doing the practical assignment you are not allowed to use the *"Lenna"* image or any other image that was used either in this book or during the presentation.

2.1 Part1 *Faces detection.*

2.1.1 original images

2.1.2 code of the scripts

```
faceDetector = vision.CascadeObjectDetector;
I = imread ('image01.jpg');
bboxes = faceDetector(I);


% Find the number of faces
```

```
faceDetector = ...
vision.CascadeObjectDetector(...
'ClassificationModel','FrontalFaceCART',...
'UseROI',true);


faceCounter=1;
for i = 1:1:length(bboxes)
    temp1 = faceDetector(I,bboxes(i,:));
    if ~isempty(temp1)
        bboxes2(faceCounter,:) = temp1;
        faceCounter = faceCounter+1;
    end
end

% Identify and frame the area of the face
IFaces = insertObjectAnnotation(I, ...
'rectangle',bboxes,'Face');
imshow (IFaces);
```

2.1.3 comments

The code detects faces in an image using a Cascade Object Detector. It initializes a detector, reads an image, detects faces in the image, and stores the bounding boxes of the detected faces. It then applies a new detector to each bounding box to identify the face and stores the bounding box of the identified face. Finally, it adds a rectangle around each detected face in the original image and displays the resulting image.

2.1.4 resulting images

2.2 Part2 *Body parts detection.*

2.2.1 original images

2.2.2 code of the scripts

```
faceDetector = vision.CascadeObjectDetector;
I = imread ('image01.jpg');
bboxes = faceDetector(I);

% Find the number of eyes
eyesDetector = ...
vision.CascadeObjectDetector(...
'ClassificationModel','EyePairSmall',...
'UseROI',true);


eyesCounter=1;
for i = 1:1:length(bboxes)
    temp2 = eyesDetector(I,bboxes(i,:));
```

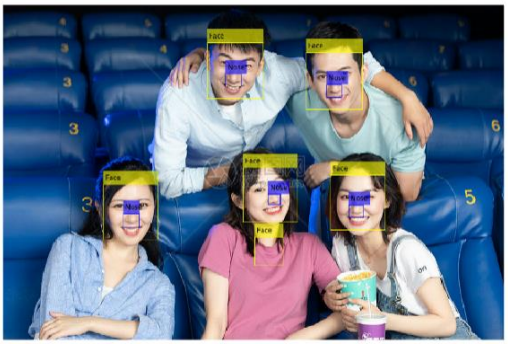
```
    if ~isempty(temp2)
        bboxes3(eyesCounter,:) = temp2;
        eyesCounter = eyesCounter+1;
    end
end

% Identify and frame the eye area
IEyes = insertObjectAnnotation (IFaces,...
 'rectangle',bboxes3,'Eyes',...
 'Color','magenta');
 imshow(IEyes)

% Find the number of noses
noseDetector = ...
vision.CascadeObjectDetector(...
'ClassificationModel','Nose',...
'UseROI',true);


noseCounter=1;
for i = 1:1:length(bboxes)
    temp = noseDetector(I,bboxes(i,:));
    if ~isempty(temp)
        bboxes4(noseCounter,:) = temp;
        noseCounter = noseCounter+1;
    end
end

% Identify and frame the area of the nose
INose = insertObjectAnnotation (IFaces,...
 'rectangle',bboxes4,'Nose',...
 'Color','blue');
 imshow(INose)
```

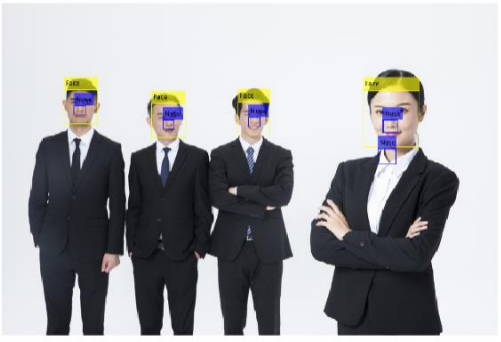
2.2.3 comments

This code uses the vision library in MATLAB to detect faces, eyes, and noses in an image. It first creates a CascadeObjectDetector for faces and applies it to the input image using imread. Then, it creates a CascadeObjectDetector for eyes and loops through the detected face bounding boxes to find the eyes within each face. It does the same for noses using a CascadeObjectDetector for noses.

Finally, it uses insertObjectAnnotation to draw rectangles around the detected eyes and noses and displays the annotated image using imshow.

2.2.4 resulting images



3. Conclusion

Faces detection and Body parts detection are very useful and advanced technologies that can provide convenience and safety for people, but at the same time they need to be constantly improved. The accuracy of recognition

will be reduced under the conditions of illumination change, attitude change, occlusion and so on. Reducing the impact of this change requires constant refinement and improvement of the code.

4. Answers to questions

1.What is the special image representation used in the Viola-Jones approach?

The special image representation used in the Viola-Jones approach is called an Integral Image. It is an intermediate representation of the original image where each pixel value is replaced by the sum of all pixel values above and to the left of it. This representation allows for very fast computation of rectangular features used in object detection.

2. What is the main advantage of Haar-like features for classifier training?

The main advantage of Haar-like features over most other features is their calculation speed. Due to the use of integral images, a Haar-like feature of any size can be calculated in constant time. This makes them particularly useful for real-time object detection.

3. Could you use Viola-Jones approach for detecting arbitrary objects and why?

Yes, the Viola-Jones approach can be used for detecting arbitrary objects. It is a machine learning object detection framework proposed in 2001 by Paul Viola and Michael Jones. Although it was primarily conceived for face detection, it can be adapted to detect other object classes as well. The algorithm is efficient and robust, achieving high precision and recall.