CS303B: AI   Lab 5

Aim: To investigate Viola-Jones object detection and camshift tracking using the Matlab Computer Vision System Tool Box.

This lab sheet is based on the Mathsworks tutorial available here: https://www.mathworks.com/help/vision/examples/face-detection-and-tracking-using-camshift.html

Face Detection and Tracking

Viola and Jones' object/face detection method is a very popular and important in many computer vision applications. In this lab, we will learn how to use Matlab to build up a simple prototype for object detection and tracking based on this method. What you will develop contains the following major steps:

1. Detect a face to track
2. Extract its hue data
3. Track the face

Step 1: Detect a Face To Track

The Computer Vision System Toolbox in Matlab is built in an object-oriented programming manner.

Before you begin tracking a face, you need to first detect it. Use `vision.CascadeObjectDetector` to estimate the location of a face in a video frame. The cascade object detector uses the Viola-Jones detection algorithm that we learned about in the lecture, and a pre-trained classification model for detection. By default, the detector is configured to detect faces, but it can be configured for other object types.

```
% Create a cascade detector object.
faceDetector = vision.CascadeObjectDetector();
% Read a video frame and run the detector.
videoFileReader = vision.VideoFileReader('visionface.avi');
videoFrame      = step(videoFileReader);
bbox = step(faceDetector, videoFrame);
% Draw the returned bounding box around the detected face.
videoOut =
insertObjectAnnotation(videoFrame,'rectangle',bbox,'Face');
figure, imshow(videoOut), title('Detected face');
```

Step 2: Extract hue data

Colour is a useful cue for face tracking since its distribution remains relatively stable as the face moves and rotates. Here we'll simply use the hue component to represent 'colour'.

```
% Extract hue by converting to the HSV color space.
[hueChannel,~,~] = rgb2hsv(videoFrame);
% Display the hue channel data and draw the bounding box
around the face.
figure, imshow(hueChannel), title('Hue channel data');
rectangle('Position',bbox(1,:),'LineWidth',2,'EdgeColor',[1 1
0])
```

Step 3: Track the Face

Now use `vision.HistogramBasedTracker` for tracking. This histogram-based tracker uses the CAMShift algorithm. In this example, the hue channel pixels are extracted from the nose region of the detected face. These pixels are used to initialize the histogram for the tracker. The example tracks the object over successive video frames using this histogram.

Please look at the complete sample code at:

https://www.mathworks.com/help/vision/examples/face-detection-and-tracking-using-camshift.html

Step 4: Detect Faces in a Video

Build a simple program based on the Viola-Jones detector to detect all faces in every frame of the video without using tracking. Display the results and compare them to the result obtained by using the face tracking results from step 3. Comment on the reasons for any differences in the results.