Lab #10

Optic Flow (17)

In this lab, we will load a video and process optic flow between each frame.

1. Take a short video with objects moving and/or the camera moving and save it to a known location on your computer. **(1)**

2. Create a video reader object using `VideoReader` to access your saved video file. **(1)**

```
vidRead = VideoReader('C:\MyFiles\myVideo.mp4');
```

3. Create a video write object using `VideoWriter` that we will use to save a processed video. **(1)**

```
vidWrite = VideoWriter('C:\MyFiles\outputVideo.avi');
```

4. Create an optic flow object using `opticalFlowFarneback`. **(1)**

```
opticFlowObj = opticalFlowFarneback;
```

5. Set up a continuous loop that will access each frame in the video being read. **(1)**

```
while hasFrame(vidRead)

end
```

6. Inside the loop, add code to read the next frame using `readFrame`. **(1)**

```
frame = readFrame(vidRead)
```

7. Resize the image frame using `imresize` (e.g., to 25%) and then convert it to grayscale. **(1)**

8. Compute the optic flow estimate for the current frame using the optic flow object (it will store the previous frame). **(1)**

```
flow = estimateFlow(opticFlowObj,frameGray);
```

9. Extract the flow field magnitude and orientation from the optic flow estimate and convert to double precision float data type. **(1)**

```
flowMagnitude = double(flow.Magnitude);
flowDirection = double(flow.Orientation);
```

10. We will combine the grayscale image and flow magnitude and direction into a colourized image with direction as hue, magnitude as saturation, and grayscale image as value. Flow direction needs to be recast from $-\pi$ to π into 0 to 1 (using `mod` and then

normalizing), and flow magnitude needs to be normalized based on some scale (e.g., 0.1). Grayscale image also needs to be converted to double type and normalized. **(1)**

```
K_mag = 0.1;
imgFlow = zeros(size(frame));
imgFlow(:,:,1) = mod(flowDirection,2*pi)/(2*pi);
imgFlow(:,:,2) = K_mag*flowMagnitude;
imgFlow(:,:,3) = double(frameGray)/255;
imgFlow = hsv2rgb(imgFlow);
```

11. Show the colourized image and add arrows plotting the optic flow estimate. Note to change the arrow colour we need to find the plot object and edit its colour field. **(1)**

```
imshow(imgFlow);
hold on
plot(flow,'DecimationFactor',[5 5],'ScaleFactor',1);
h = findobj(gca,'Type','quiver');
h.Color = 'w';
```

12. Capture the current plotted frame using `getframe` and write to the video file using `writeVideo`. **(1)**

```
frameOut = getframe(gca);
writeVideo(vidWrite,frameOut);
```

13. At the end, outside of the loop, close the video writer object. **(1)**

```
close(writeVideo);
```

14. Review the saved video. **Discuss the resulting flow analysis of your video. Was it was expected? Can you see areas in the video flow analysis is incorrect? (5)**