Individual assignment 4:
From basic computer vision towards advanced object detection

The goal of this individual assignment is to apply some basic computer vision techniques to detect an object and produce a one minute video for submission. You will have to submit this video output and your code by **Thursday 19 October, 2023 23:59 (CET)**.

1 Introduction

In this session, you are the director, scenario writer, producer and at the same time actor of a new short film of **1 minute**. Your piece of art will be assessed (score is dependent on technical performance and following the assignment guidelines. Your first step as a producer is the pre-production, **we strongly advise you to** :

1. first read through this assignment carefully, think about the story line (all image processing or computer vision techniques that you are going to apply).
2. In the production step, you do the recording (use your phone/webcam), note that the footage should contain at least some object(s).
3. In the post-production you add effects by applying computer vision or image processing methods, show off some techniques and finally stitch all scenes together into a one minute video film. **Provide subtitles to describe which techniques you apply where, this helps to draw the attention of the jury members to the important bits.**

2 Requirements

Your final film should be structured as followed. It consists of **4 parts**.

All the listed techniques are considered as compulsory tasks (except Freestyle part). Skipping any of them will cause reductions in the grade.

- **Part 1: from 0s to 10s**: Apply basic image processing techniques to provide special effects from spatial domain **(total 3 points)**

topics	Required method	points
spatial filters	smoothing or blurring to reduce noise (choose your own kernel/mask)	0.5
	Sharpen the image	0.5
Edge detection	Sobel edge operator	1
	Canny edge operator, add parameters of canny in subtitle	1

- **Part 2: from 10s to 20s**: Apply Fourier Transform to provide the special effects from frequency domain **(total 2 points)**

topics	Required method	points
Fourier Transform	Show DFT spectrum of current image frame	0.5
	Apply low pass filter, indicate in subtitle which low pass filter, show the Inversed FFT result in spatial domain	0.5
	Apply **high** pass filter, indicate in subtitle which high pass filter, show the Inversed FFT result in spatial domain	0.5
	Apply **band** pass filter, indicate in subtitle which band pass filter, show the Inversed FFT result in spatial domain	0.5

- **Part 3: from 20s to 40s**: Apply learned object detection methods **(Total 3 points)**

topics	Required method	points
Template matching	Define your template and apply template matching to the video frame. Tip: to choose a template with rich features.	0.5
	show the found match pattern with lines/circles at least lasting for 5 seconds	1
Optical flow tracking	Apply optical flow to the video frames, show the result with arrows to indicate the optical flow results.	1.5

- **Part 4: 40s-60s**: **Freestyle** part!! Do whatever it takes to end with a bang! (**total 2 points**)
    - It is encouraged to perform some techniques related to object tracking or detection. E.g. tracking an object with different methods or with image stitching. In the end, you are free to choose anything to make it COOL!
    - The grade will be evaluated based on the image processing and computer vision techniques you applied and their effects on the video. creativity is the main factor here. To use a simple method for realising a surprising effect could be one of the examples.

If you want to use Pyhton. You need to use Python and the OpenCV library (or other libraries, e.g. scikit-image) for the implementation. The actual video capture can be done using standard video recording tools (e.g. your smartphone or PC video recording software) and the video processing can be done with OpenCV. You can process the video online and simply process it frame by frame or experiment using temporal information. Using OpenCV in this repository: https://github.com/gourie/opencv_video. Also available on the Canvas page.

If you want to use Matlab, many resources online, feel free to use. Here is one example. https://nl.mathworks.com/help/vision/ug/motion-based-multiple-object-tracking.html

Lastly, while most of the computation will be rather efficient, it can be a good idea to **downsample your video right from the beginning**. You will anyway have to make sure your video **smaller than 30 MB** before you upload it on Canvas. It's up to you now, good luck!

3 Submission

Submit the video and your code separately on Canvas by **Thursday 19 October, 2023 23:59 (CET)**..
To do so, first make sure to compress and upload the video under MPEG-4 (MP4) format. Then, also upload all your code as a single compressed ZIP file. The video you upload preferably is (or should be) smaller than 30 MB in size. Remember that you can choose to downsample your video right from the start, or work on full resolution and downsample it just before compressing and uploading it on Canvas. **Submissions with other compression than ZIP for the code and/or with videos larger than 30 MB and/or in a different format than MP4 will not be reviewed!**

Submission checklist:

1. **Original video (downsampled)**
2. **produced video (downsampled)**
3. **your code(s) (check relative path) (make sure it can run correctly on other pc or laptop. Test with your classmate before submission)**