Laboratory on Change detection and Tracking Computer Vision

University of Genoa - Robotics Engineering Master's degree

Ayan Mazhitov, Madi Nurmanov
mazhitovayan@gmail.com, me.nurmanov@gmail.com

January 4, 2022

Abstract

This report is dedicated for the explanation of the Computer Vision laboratory work on Change detection and Tracking. As the name suggests, the main focus of the tasks is based on a sequence of images, from which information on the scene and its elements can be drawn. This is useful for change detection and tracking purposes in various computer vision applications.

1 Introduction

Change detection and Tracking are one of the main tools that are used in computer vision pipeline. They can be deployed in many examples: security surveillance systems, smart-houses and etc.

In this laboratory work, different techniques for change detection are used. The key aspect is in the frames used as a reference for the background. It is important step, as it tries to minimize the influence of constant changes in the scene and variations of illumination conditions.

First approach uses simple average between two empty frames. Second approach takes the running average as a basis for the background model. The third technique is fundamentally different in comparison with the first two, as it incorporates multi-modal configuration for the background basis frame.

The last part of this laboratory work focuses on tracking. The guidelines for the design of algorithm were provided, so that meaningful results could be obtained.

2 Methods

2.1 Change detection: two empty background reference frames

Reference frame is assumed to be a background of the scene without major changes in its constitution. In other words, there is no motion, only static image of the environment is present. This can be done by averaging N frames and the result can be used for subtraction with the stream of images to obtain the motion as a difference between static and "dynamic" pictures. The implementations simply sources two "empty" images to MATLAB taking the average of two for the background model. After subtraction of stream frames with the reference frame, the resulting image should represent the change which is correlated to the movement registered on the video.

2.2 Change detection: running average

Another approach is to use a running average, as it is more adaptable for the changes in the environment. For instance, light conditions of the scene are subject to deviation due to various reasons. Not only this, but there might be single-time changes of the scene, which should not be always considered as a movement and therefore require exclusion from the region of interest. Using a constantly updated base frame at each time instance solves this problem.

2.3 Change detection: running average - hyper parameters

Running average approach can be tuned by the parameters tau and alpha. In this part of the laboratory work, influence of these parameters is subject for investigation. Tau stands for the threshold and alpha is a coefficient proportional to the rate of change.

2.4 Tracking

The above parts of the laboratory work create a basis for a more informative application of the computer vision algorithms. It is possible to track the objects moving in the scene by associating the connected components between the frames in a sequence (i.e object on the new scene with object on the previous scene). Based on that information, the trajectory and its type can be retrieved, producing meaningful results that can be further processed upon request.

3 Results

3.1 Change detection: two empty background reference frames

The system has proven to be efficient, yet there are situations when noisiness of the output data can lead to false detection of changes. This is due to the fact that reference frame is restricted only for two empty frames. For this experiment the value of *tau* is set to 20.

3.2 Change detection: running average

To avoid the problem of false detection faced in the previous implementation this approach considers dynamic reference when creating a background model. This allows to considers permanent changes, as well as it smooths the influence of lighting change. For this task, the values of *alpha* and *tau* were set to 0.9 and 25 respectively.

Figure 1: Results obtained in task 2

3.3 Change detection: running average - tuning hyper parameters

Tau influences the balance between black and white colours while alpha minimizes the amount of false change detection when decreased, as noises have less influence on the final output, since lower value of alpha filters out too rapid changes. Here, the value of *tau* is set to 20 and *alpha* to 0.01 in order to compare the results with previous part of the experiment. To do so, it is required to compare Fig. 1 and Fig. 2

Figure 2: Results obtained in task 3

As it can be seen, the features of each object after tuning the hyper-parameters are more distinct and can be separated for different blobs for a further examination.

3.4 Tracking

After finding the proper hyper parameters suitable for the scene, the tracking algorithm was designed. First using the techniques implemented in the previous parts of this work, we could differentiate moving from still pixels. After that, we used built-in MatLab function, such as ***imfill() bwconncomp()*** to unite neighbourhood pixels in one unique object. After that we decided to remove the objects that were too smal to be considered as moving objects, because that could just be illumination inaccuracies. By defining 500 as an area threshold, we remove undesired objects. After that, we calculate the center of each object and insert it into dictionary variable. In the consecutive frames, we compare the newly found moving objects' center with known centers in our dictionary variable. If the newly found moving object's center is close enough to already existing in the dictionary, we can decide that it is indeed the same object. Thus, we update the history of this objects. Assigning the specific IDs for the each new object identified in the scene, it was decided to put the objects in the bounding boxes as shown in Fig. 3.

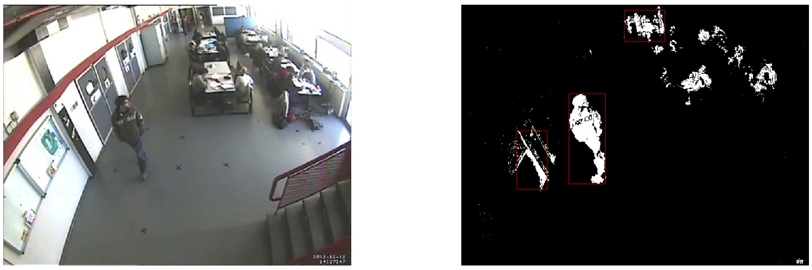

Figure 3: Bounding boxes for Tracking

Moreover, the center's of these bounding boxes are marked on each frame to create a trajectory allowing to track the movement across the scene from the stream. At the end, we visualize the trajectory by plotting the line between each point in the history. This is demonstrated in Fig. 4.

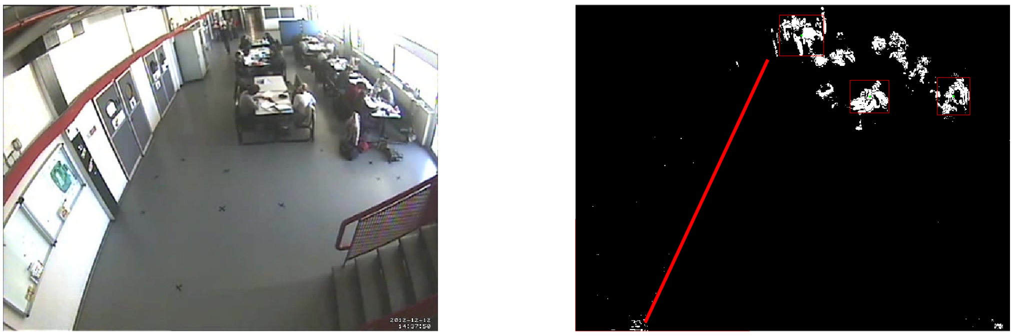

Figure 4: Tracking the Objects in Bounding Boxes

4 Conclusion

This laboratory report covered the implementation of sequential image processing techniques for change detection and tracking purposes. Firstly, the performance of change detection system based on two empty frames was examined. Afterwards, it was improved by introduction of running-average as a background model, since it is more flexible in mitigation of static changes and lighting condition's influence of the frame. Another step in improving the change detection system is to tune the hyper parameters *alpha* and *tau* for improving the performance of the system with given input. It was practically proven, that running-average outperforms set of static empty images as a background model, while decreasing the value of *alpha* reduces the noise in binary map, essentially allowing to focus on the main blobs of the considered scene. The last part of this laboratory work considered the implementation of tracking system. It was possible to derive the trajectory of the target objects, meaning that the space-time parameters were successfully recovered from the input stream.