International Journal of Enhanced Research in Science, Technology & Engineering
ISSN: 2319-7463, Vol. 7 Issue 7, July-2018, Impact Factor: 4.059

Detecting Car Accidents using optical flow

Ismail El-Sayad[1], Samih Abdul-Nabi[2], Hussien Kassem[3],
Zahraa Bader el dine[4], Myram Abdallah[5]

[12345]*Lebanese International University, Beirut, Lebanon*

ABSTRACT

Nowadays, our life is highly dependent on the technology that engineers and scientists have developed during the past years. For instance, it is now possible to detect an event such as car accident using automation systems and computers. Video surveillance event detection is an automated system that uses an ontology to directly detect any abnormal event happening at the right time. Such system decreases time, effort and money spent on bigger numbers of security guards keeping their eyes on different screens all the time. Our system can detect car accident at the specific time and the exact place by sensing and recognizing motion.

Keywords: **Optical Flow, Motion Detection, HOM, SIFT, Event detection.**

I. INTRODUCTION

Videos, photo shooting, snapping, house partying, updating the Facebook picture, posting a new one on Instagram, taking an id picture, picturing a nice view, a nice gathering are all things that we do every moment of our days, they all have in common a single point, cameras. A camera isn't only picturing a frame, it's picturing a moment, a moment of sadness, happiness, it may be full of emotions, events, and others. That's why the complications of cameras and their tolerance have been a very big deal to the engineers in the last few decades.

They have been studying billions of ways to increase their specifications and making them more professional in terms of colors, sharpness, and accuracy of objects. In this case of studying, we are mostly interested in the once concerning surveillance as shown in Figure 1. Surveillance camera is a major part of our security systems nowadays, including malls, highways, business companies and others can no more operate without security systems where wherever we are fully conscious that our safety is achieved by a very big security system covered and managed by a huge number of cameras each one operating in a different location to cover the whole globe that we are in no matter how huge it is.

Figure 1: Employee keeping his eye on the screens to detect any weird movement.

Mainly to avoid robbing, riots, and terrorist actions these surveillance cameras are found, where if they didn't help to prevent these actions they will help in discovering the people behind them later on. With its tolerance and development surveillance cameras nowadays have reached a point where they could operate without an eye over them where a weird event by itself could be detected using a system that is designed for this issue without the interference of human.

In any mall, any hotel, airports, streets, and most place we walk right now, we know that every single movement is detected by security surveillance cameras. It is an obliged thing for any business to have a security room full of screens for security with at least one or two employees to monitor any weird movement and detect it, and whenever the business grows bigger the number of security employees grows bigger as shown in **Error! Reference source not found.**.

Figure 2: Huge number of screens is needed in big companies

However, sometimes the security guard can't really detect the events happening maybe because he's busy with a different screen, so sometimes he may miss very important actions happening and he may miss the control over the event. Actually, we need something that decreases time, effort and money spent on bigger numbers of a security guard. Thus a system should be proposed that can detect the events at the specific time of happening by sensing and recognizing motion through observing weird movements and actions.

We cannot say that a motion is a sufficient indication of a threatening or interesting action. There is a big amount of objects that are moving in many natural scenes, analyzing their trajectories and their interaction with the features of the scene, help in classifying and recognizing the important events based on specific criterion. Consequently, the recognition and prediction of human activities from videos pose a challenge for the present approaches to video event detection because of the difficulty in segmenting the actor from the background imposed by any distracting motion coming from other objects.

To achieve our goals; the recognition and prediction of weird event from videos, we propose an approach based on identifying and characterizing in a valuable way the areas in motion, then modeling of the retrieved descriptors using a proposed ontology that integrates both scene and system information to sum up with a final decision about the existence of an event (crowded people - accident ….).[1]

The recognition and prediction of people activities from videos are a major concern in the field of computer vision. The main objective of our thesis is to propose an algorithm that analyzes each scene of a video to recognize a specific event. This problem is also called VCA or video content analysis. This analysis is performed in outdoor or indoor environments using simple surveillance cameras.

Actually, an approach based on three levels of analysis is proposed. The first level is the detection of low-level descriptors retrieved from the images of the video (e.g. areas in motion). The second level retrieves descriptors for modeling human behavior (e.g. average speed and direction of movement). The top level uses the descriptors of the intermediate step to provide users with concrete results on the analysis of behavior (e.g. this person is running, that one is walking, etc.).

II. PROPOSED METHODOLOGY

We chose to work on optical flow specifically working with histograms of optical flow orientation within each range of magnitudes. Histogram of optical flow orientation and magnitude (HOFM) [1,2,3] is a spatiotemporal feature descriptor used to describe normal patterns and it encodes both magnitude and orientation of the optical flow separately into histograms. We can employ a simple nearest neighbor search to identify whether a given unknown pattern should be classified as an anomalous event. HOFM captures not only the orientation, but also the magnitude of the flow vectors, which provide information regarding the velocity of the moving objects and improves considerably the representation of the normal events. The main idea for the detection of anomalous event is to search for some pattern which is similar to the incoming pattern.

Nearest neighbor search. Anomalous event pattern is represented by case of point A and a normal event pattern by case of point B. If the incoming pattern is likely enough to some of the known patterns, then it is considered as a normal pattern (case of point B), otherwise, if the incoming pattern is not close to any learned patterns, it will be considered as an anomalous event.

Our proposed design for detecting a car accident is shown in figure 3.

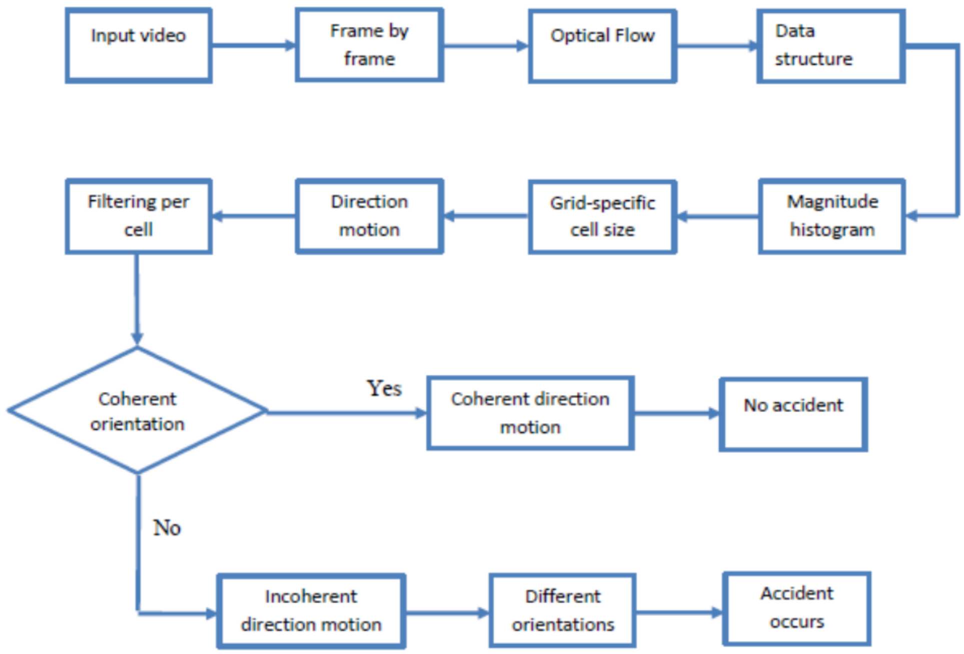


Figure 3: Proposed Design

As steps for our summarized design we starts by:

1- Applying optical flow using open-cv on a video to get an orientations and magnitudes of vectors for each frame in the video.

Figure 4: Detecting Optical flow

2- Apply a Grid that divide each frame into cells based on the location of each interest point. [4,5]

Figure 5: Segmentation of the frame into cells

3- The size of each cell in the grid depends on the generated histograms
4- The hot cells are selected and are the cells for the street, so we can process only these cells that we are interested in.

Figure 6: Hot Cells

5- Apply filtering for the movement per cell by calculating the histogram of orientation within each range of magnitudes for example, suppose range 1 consisting of vectors having magnitudes from 0 to 20, range 2 from 21 to 40 and so on, there are 3 vectors going to the left in range 1 and 1 in range 2.

The presence of movement that is not coherent with the neighbor movement is an indication of abnormal event. And a lot of orientations will indicate incoherent motion. The accident will generate abnormal information on the motion direction. [6,7,8]

3. IMPLEMENTATION AND SIMULATION RESULTS

The design consideration of our system was the programming language that suits best our simulation, while various options such as MATLAB and JAVA were employed, our chosen one is C++ using opencv due to various reasons concerning the availability of libraries for image processing.

We have considered two scenarios. The first scenario is normal movement scenario as seen in Figure 7. The second one is the car accident scenario as shown in Figure 8.

International Journal of Enhanced Research in Science, Technology & Engineering
ISSN: 2319-7463, Vol. 7 Issue 7, July-2018, Impact Factor: 4.059

Figure 7: Normal Movement scenario

Figure 8: Car accident Scenario

We build histograms of magnitude as shown in figures 9 and 12. These histograms represents the number of optical flow vectors in each range of magnitudes within each cell for the two scenarios. [9,10].

Based on our data set we chose the ranges as follow:

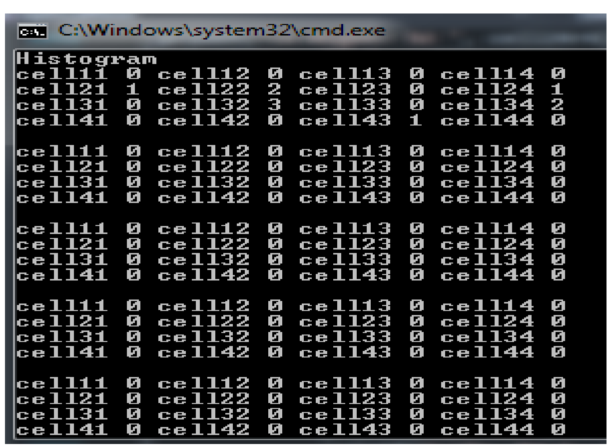
Range 1: magnitude less than or equals to 2.

Range 2: magnitude between 2 and 60.

Range 3: magnitudes between 60 and 100.

Range 4: magnitudes between 100 and 250.

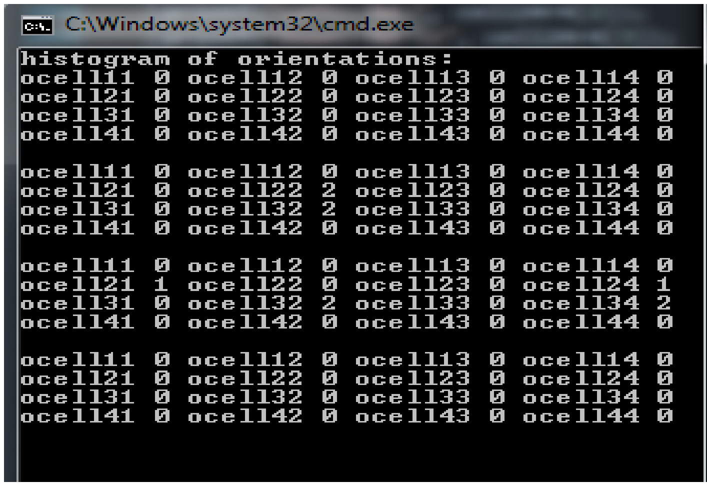
Range 5: magnitudes greater than 250.



```
C:\Windows\system32\cmd.exe
Histogram
cell11 0 cell12 0 cell13 0 cell14 0
cell21 1 cell22 2 cell23 0 cell24 1
cell31 0 cell32 3 cell33 0 cell34 2
cell41 0 cell42 0 cell43 1 cell44 0

cell11 0 cell12 0 cell13 0 cell14 0
cell21 0 cell22 0 cell23 0 cell24 0
cell31 0 cell32 0 cell33 0 cell34 0
cell41 0 cell42 0 cell43 0 cell44 0

cell11 0 cell12 0 cell13 0 cell14 0
cell21 0 cell22 0 cell23 0 cell24 0
cell31 0 cell32 0 cell33 0 cell34 0
cell41 0 cell42 0 cell43 0 cell44 0

cell11 0 cell12 0 cell13 0 cell14 0
cell21 0 cell22 0 cell23 0 cell24 0
cell31 0 cell32 0 cell33 0 cell34 0
cell41 0 cell42 0 cell43 0 cell44 0

cell11 0 cell12 0 cell13 0 cell14 0
cell21 0 cell22 0 cell23 0 cell24 0
cell31 0 cell32 0 cell33 0 cell34 0
cell41 0 cell42 0 cell43 0 cell44 0
```

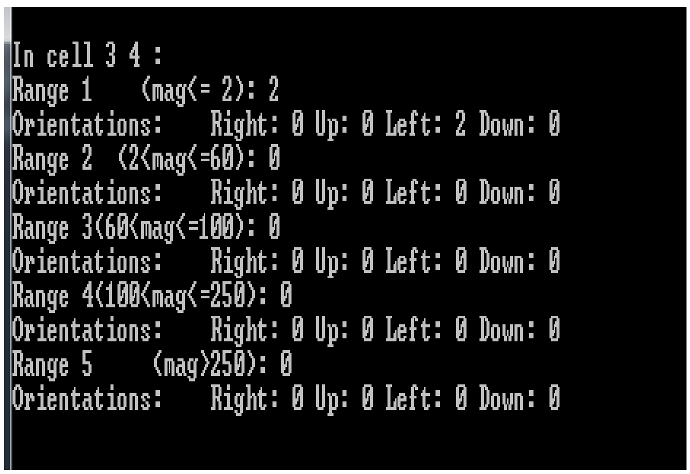
Figure 9: Normal movement histogram of magnitudes

International Journal of Enhanced Research in Science, Technology & Engineering
ISSN: 2319-7463, Vol. 7 Issue 7, July-2018, Impact Factor: 4.059



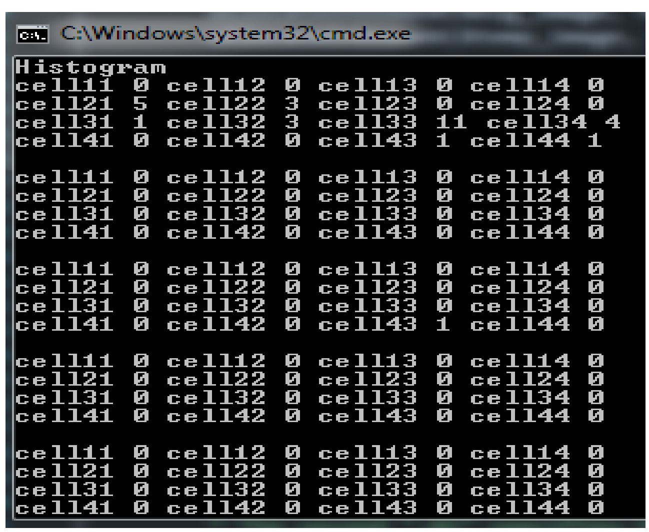
```
C:\Windows\system32\cmd.exe
histogram of orientations:
ocell11 0 ocell12 0 ocell13 0 ocell14 0
ocell21 0 ocell22 0 ocell23 0 ocell24 0
ocell31 0 ocell32 0 ocell33 0 ocell34 0
ocell41 0 ocell42 0 ocell43 0 ocell44 0

ocell11 0 ocell12 0 ocell13 0 ocell14 0
ocell21 0 ocell22 2 ocell23 0 ocell24 0
ocell31 0 ocell32 2 ocell33 0 ocell34 0
ocell41 0 ocell42 0 ocell43 0 ocell44 0

ocell11 0 ocell12 0 ocell13 0 ocell14 0
ocell21 1 ocell22 0 ocell23 0 ocell24 1
ocell31 0 ocell32 2 ocell33 0 ocell34 2
ocell41 0 ocell42 0 ocell43 0 ocell44 0

ocell11 0 ocell12 0 ocell13 0 ocell14 0
ocell21 0 ocell22 0 ocell23 0 ocell24 0
ocell31 0 ocell32 0 ocell33 0 ocell34 0
ocell41 0 ocell42 0 ocell43 0 ocell44 0
```

Figure 10: Normal movement histogram of orientations



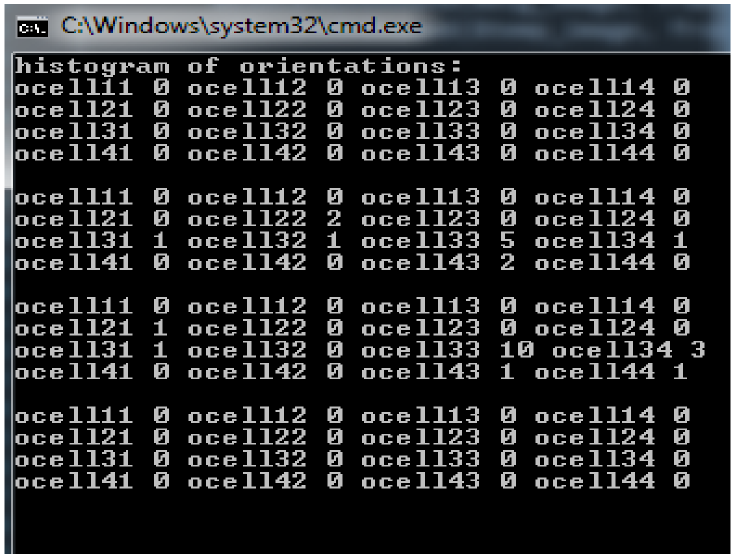
```
In cell 3 4 :
Range 1    (mag<= 2): 2
Orientations:    Right: 0 Up: 0 Left: 2 Down: 0
Range 2  (2<mag<=60): 0
Orientations:    Right: 0 Up: 0 Left: 0 Down: 0
Range 3(60<mag<=100): 0
Orientations:    Right: 0 Up: 0 Left: 0 Down: 0
Range 4(100<mag<=250): 0
Orientations:    Right: 0 Up: 0 Left: 0 Down: 0
Range 5     (mag>250): 0
Orientations:    Right: 0 Up: 0 Left: 0 Down: 0
```

Figure 8: Normal movement histogram of magnitudes and orientations



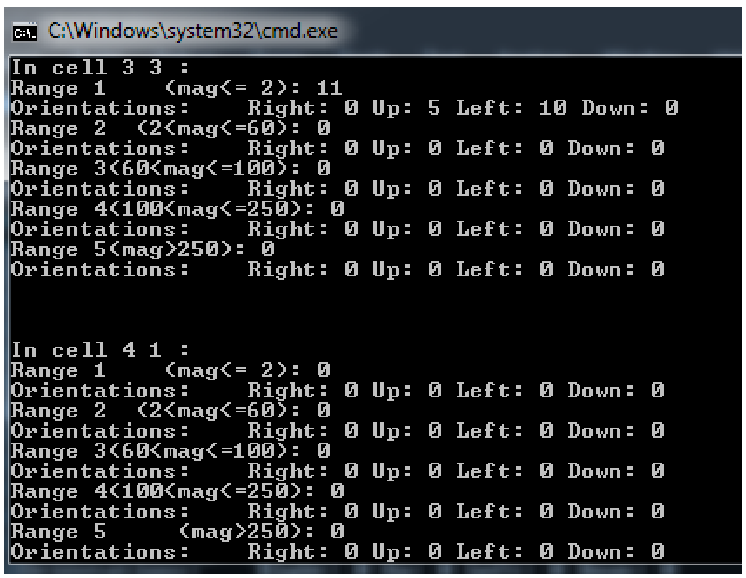
```
C:\Windows\system32\cmd.exe
Histogram
cell11 0 cell12 0 cell13 0 cell14 0
cell21 5 cell22 3 cell23 0 cell24 0
cell31 1 cell32 3 cell33 11 cell34 4
cell41 0 cell42 0 cell43 1 cell44 1

cell11 0 cell12 0 cell13 0 cell14 0
cell21 0 cell22 0 cell23 0 cell24 0
cell31 0 cell32 0 cell33 0 cell34 0
cell41 0 cell42 0 cell43 0 cell44 0

cell11 0 cell12 0 cell13 0 cell14 0
cell21 0 cell22 0 cell23 0 cell24 0
cell31 0 cell32 0 cell33 0 cell34 0
cell41 0 cell42 0 cell43 1 cell44 0

cell11 0 cell12 0 cell13 0 cell14 0
cell21 0 cell22 0 cell23 0 cell24 0
cell31 0 cell32 0 cell33 0 cell34 0
cell41 0 cell42 0 cell43 0 cell44 0

cell11 0 cell12 0 cell13 0 cell14 0
cell21 0 cell22 0 cell23 0 cell24 0
cell31 0 cell32 0 cell33 0 cell34 0
cell41 0 cell42 0 cell43 0 cell44 0
```

Figure 9: Histogram of magnitudes in case of accident



```
C:\Windows\system32\cmd.exe
histogram of orientations:
ocell11 0 ocell12 0 ocell13 0 ocell14 0
ocell21 0 ocell22 0 ocell23 0 ocell24 0
ocell31 0 ocell32 0 ocell33 0 ocell34 0
ocell41 0 ocell42 0 ocell43 0 ocell44 0

ocell11 0 ocell12 0 ocell13 0 ocell14 0
ocell21 0 ocell22 2 ocell23 0 ocell24 0
ocell31 1 ocell32 1 ocell33 5 ocell34 1
ocell41 0 ocell42 0 ocell43 2 ocell44 0

ocell11 0 ocell12 0 ocell13 0 ocell14 0
ocell21 1 ocell22 0 ocell23 0 ocell24 0
ocell31 1 ocell32 0 ocell33 10 ocell34 3
ocell41 0 ocell42 0 ocell43 1 ocell44 1

ocell11 0 ocell12 0 ocell13 0 ocell14 0
ocell21 0 ocell22 0 ocell23 0 ocell24 0
ocell31 0 ocell32 0 ocell33 0 ocell34 0
ocell41 0 ocell42 0 ocell43 0 ocell44 0
```

Figure 10: Histogram of orientations in case of accident



```
C:\Windows\system32\cmd.exe
In cell 3 3 :
Range 1     (mag<= 2): 11
Orientations:    Right: 0 Up: 5 Left: 10 Down: 0
Range 2  (2<mag<=60): 0
Orientations:    Right: 0 Up: 0 Left: 0 Down: 0
Range 3(60<mag<=100): 0
Orientations:    Right: 0 Up: 0 Left: 0 Down: 0
Range 4(100<mag<=250): 0
Orientations:    Right: 0 Up: 0 Left: 0 Down: 0
Range 5(mag>250): 0
Orientations:    Right: 0 Up: 0 Left: 0 Down: 0


In cell 4 1 :
Range 1     (mag<= 2): 0
Orientations:    Right: 0 Up: 0 Left: 0 Down: 0
Range 2  (2<mag<=60): 0
Orientations:    Right: 0 Up: 0 Left: 0 Down: 0
Range 3(60<mag<=100): 0
Orientations:    Right: 0 Up: 0 Left: 0 Down: 0
Range 4(100<mag<=250): 0
Orientations:    Right: 0 Up: 0 Left: 0 Down: 0
Range 5      (mag>250): 0
Orientations:    Right: 0 Up: 0 Left: 0 Down: 0
```

Figure 11: Histogram of orientations and magnitudes in case of accident

We have also built histograms of orientations as shown in Figures 10 and 13 that consist of 4 sets and are arranged as follow:

Set 1: number of vectors going to the right direction in each cell.
Set 2: number of vectors going to the up direction in each cell.
Set 3: number of vectors going to the left direction in each cell.
Set 4: number of vectors going to the down direction in each cell.

Note that a single vector may have 2 directions in the horizontal or vertical plane (right/left and up/down).

Finally we construct histograms of magnitudes and orientations figure (figures: 11 and 14) shows within each range of magnitudes the number of vectors in each direction in a specific cell.

In figure 15 where the accident occurs we can see that the number of vectors becomes 11 in cell 3-3; 5 of them to the up direction and 10 to the left (see figure 27) which is abnormal.

Figure 15: Car accident detection

So we chose a threshold of 9 to notify the system about a car accident where if there is more than 9 vector directing to the left in a specific cell with range less than 2 we can say there is an accident happening at that position or cell.

CONCLUSION

We proposed an algorithm that analyzes each scene of a video to recognize a specific event. Video content analysis is performed in outdoor or indoor environments using simple surveillance cameras, in our case we captured a video of the street and compared the results obtained in a normal case and in the case of an accident and based on the results we built our ontology that can be used to detect any similar event.

The approach is based on different levels of analysis. The first one is the detection of low-level descriptors retrieved from video frames while the second retrieves descriptors to be used by the top level to provide us with the concrete results.

The designed system is suitable for all people looking for safety and recognition of such accidents at right time. Moreover, it is scalable since it can be extended to include many more kinds of events using more different data sets.

REFERENCES

[1]. Liu, Y., Zhang, D., Lu, G., & Ma, W. Y. (2007). A survey of content-based image retrieval with high-level semantics. Pattern recognition, 40(1), 262-282.
[2]. Sykora, P., Kamencay, P., & Hudec, R. (2014). Comparison of SIFT and SURF methods for use on hand gesture recognition based on depth map. AASRI Procedia, 9, 19-24.
[3]. Benabbas, Y., Ihaddadene, N., & Djeraba, C. (2011). Motion pattern extraction and event detection for automatic visual surveillance. Journal on Image and Video Processing, 2011, 7.
[4]. SanMiguel, J. C., Martinez, J. M., & Garcia, Á. (2009, September). An ontology for event detection and its application in surveillance video. In Advanced Video and Signal Based Surveillance, 2009. AVSS'09. Sixth IEEE International Conference on (pp. 220-225). IEEE.
[5]. Farnebäck, G. (2003, June). Two-frame motion estimation based on polynomial expansion. In Scandinavian conference on Image analysis (pp. 363-370). Springer, Berlin, Heidelberg.
[6]. Allaert, B., Bilasco, I. M., & Djeraba, C. (2017, February). Consistent Optical Flow Maps for Full and Micro Facial Expression Recognition. In VISIGRAPP (5: VISAPP) (pp. 235-242).
[7]. Wang, H., Kläser, A., Schmid, C., & Liu, C. L. (2013). Dense trajectories and motion boundary descriptors for action recognition. International journal of computer vision, 103(1), 60-79.
[8]. Revaud, J., Weinzaepfel, P., Harchaoui, Z., & Schmid, C. (2015). Epicflow: Edge-preserving interpolation of correspondences for optical flow. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1164-1172).
[9]. Colque, R. V. H. M., Caetano, C., de Andrade, M. T. L., & Schwartz, W. R. (2017). Histograms of optical flow orientation and magnitude and entropy to detect anomalous events in videos. IEEE Transactions on Circuits and Systems for Video Technology, 27(3), 673-682.
[10]. Tani, M. Y. K., Lablack, A., Ghomari, A., & Bilasco, I. M. (2014, September). Events detection using a video-surveillance ontology and a rule-based approach. In European Conference on Computer Vision (pp. 299-308). Springer, Cham. approach,"