

Real Time Detection of Moving Object based on Optical Flow Method

Lekshmi B

Applied Electronics and Instrumentation
Younus College Of Engineering And Technology
Kollam,India

Safuvan T

Assistant Professor
Younus College Of Engineering And Technology
Kollam,India

Abstract—Motion detection is considered to be a very important task in image processing. Optical flow is one way of detecting moving objects. Optical flow cannot be computed locally, since only one independent measurement is available from the image sequence at a point, while the flow velocity has two components. In this project the method used for finding the optical flow is assuming that the apparent velocity of the brightness pattern varies smoothly almost everywhere in the image. This technique is later used in developing software for motion detection methods which has the capability to carry out four types of motion detection. The motion detection software presented in this project also capable to highlight motion region, count motion level as well as counting object numbers. A wide variety of objects such as vehicles and human from video streams can be recognized by applying optical flow technique.

Keywords—Background modeling, Motion detection, Optical flow, Velocity smoothness constant, motion trajectories.

I. INTRODUCTION

Detecting moving objects has been widely applied in computer vision, so it attracts intensive attention from researchers in the area of image processing. However, because the surroundings in the real-world videos are often quite articulated or even non-rigid, moving object detection is still a challenging problem that needs to be further addressed. As indicated in [1], there are mainly three factors that make it more difficult to detect moving objects from videos. (1) the presence of complex background, e.g., dynamic background with swaying trees, (2) camera motion, e.g., tripod vibration, (3) requiring prior knowledge, e.g., training data for modelling the background. Furthermore, most existing moving object detection algorithms are not intelligent or robust enough for that they need user interaction or experiential parameter tuning. To tackle with these problems, we propose novel motion detection method based on optical flow. Optical flow is the distribution of apparent velocities of movement of brightness patterns in an image. Optical flow can arise from relative motion of moving objects and the viewer [1, 2]. Consequently, optical flow can give important information about the spatial arrangement of the objects viewed and the rate of change of this arrangement [3]. Discontinuities in the optical flow can help in segmenting images into regions that correspond to different objects [4]. The term optical flow is also used by robotics, encompassing related techniques from image processing and control of navigation including motion detection, object segmentation, time-to-contact information,

focus of expansion calculations, luminance, motion compensated encoding, and stereo disparity measurement.

Motion estimation and video compression have developed on the basis of optical flow research. The optical flow field is superficially similar to a dense motion field derived from the techniques of motion estimation. Optical flow is the study of not only the determination of the optical flow field itself, but also of its use in estimating the three-dimensional nature and structure of the scene, as well as the 3D motion of objects and the observer relative to the scene, most of these images using the Image Jacobian.

Optical flow was used by robotics researchers in many areas such as: object detection and tracking, image dominant plane extraction, movement detection, robot navigation and visual odometry. Optical flow information has been recognized as being useful for controlling micro air vehicles.

The application of this method includes the problem of inferring not only the motion of the observer and objects in the scene, but also the structure of objects and the environment. Since awareness of motion and the generation of mental maps of the structure of our environment are critical components of animal (and human) vision, the conversion of this innate ability to a computer capability is similarly crucial in the field of machine vision.

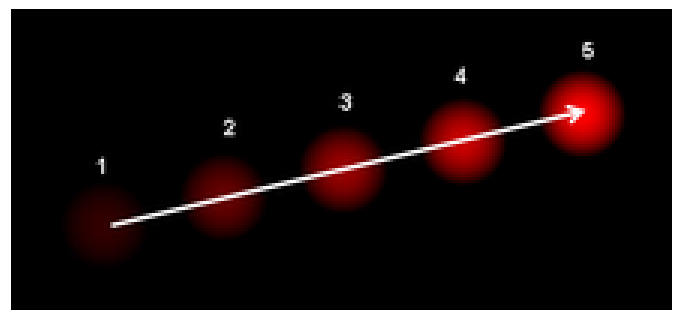


Fig. 1. The optical flow vector of a moving object in a video sequence

Consider a five-frame clip of a ball moving from the bottom left of a field of vision, to the top right. Motion estimation techniques can determine that on a two dimensional plane the ball is moving up and to the right and vectors describing this motion can be extracted from the sequence of frames. For the purposes of video compression (e.g., MPEG), the sequence is now described as well as it needs to be. However, in the field of machine vision, the

question of whether the ball is moving to the right or if the observer is moving to the left is unknowable yet critical information. Not even if a static, patterned background were present in the five frames, could we confidently state that the ball was moving to the right, because the pattern might have an infinite distance to the observer. The optical flow describes the direction and time pixels in a time sequence of two consequent dimensional velocity vector, carrying direction and the velocity of motion is assigned to each pixel in a given place of the picture.

II. OPTICAL FLOW IN MOTION ANALYSIS

Optical flow gives a description of motion and can be a valuable contribution to image interpretation even if no quantitative parameters are obtained from motion analysis. Optical flow can be used to study a large variety of motions—moving observer and static objects, static observer and moving objects, or both moving. Optical flow analysis does not result in motion trajectories instead, more general motion properties are detected that can significantly increase the reliability of complex dynamic image analysis [5]. Motion, as it appears in dynamic images, is usually some combination of four basic elements:

- i) Translation at constant distance from the observer.
- ii) Translation in depth relative to the observer.
- iii) Rotation at constant distance about the view axis.
- vi) Rotation of a planar object perpendicular to the view axis.

Optical-flow based motion analysis can recognize these basic elements by applying a few relatively simple operators to the flow [6]. Motion form recognition is based on the following facts:

- a) Translation at constant distance is represented as a set of parallel motion vectors.
- b) Translation in depth forms a set of vectors having a common focus of expansion.
- c) Rotation at constant distance results in a set of concentric motion vectors.
- d) Rotation perpendicular to the view axis forms one or more sets of vectors starting from straight line segments. Exact determination of rotation axes and translation trajectories can be computed, but with a significant increase in difficulty of analysis.

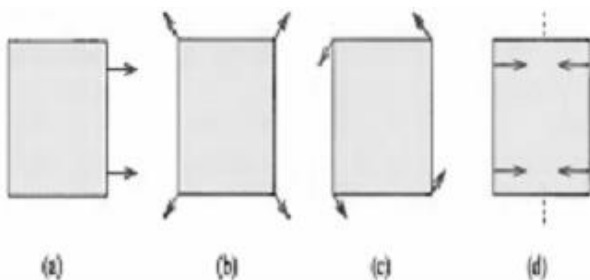


Fig. 2. Motion form recognition, (a) Translation at constant distance, (b) Translation in depth, (c) Rotation at constant distance, (d) Planar object rotation perpendicular to the view

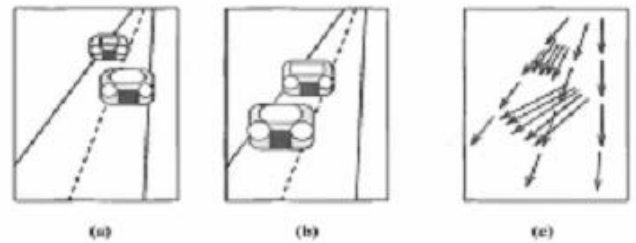


Fig. 3. The Focus of expansion, (a) Time t1 (b) Time t2. (c) Optical flow

A major interests of motion analysis is to estimate 3D motion. The motion analysis tasks could be roughly categorized into three different settings: 2D-2D, 2D-3D, and 3D-3D, depending on the correspondences. The 3D-3D problem is to calculate 3D motion based on a set of 3D correspondences. But generally, direct 3D data are difficult to obtain. To ease the problem, we can assume 2D-3D correspondences. The 2D-3D problem is to determine 3D motion based on the correspondence between 3D model and 2D image projections. 3D model-based analysis is one of such examples. Without using any 3D models, 2D-2D analysis only assume the correspondences between 2D image projections but aim at calculating 3D motion from such 2D correspondences. A critical but difficult problem for motion analysis, obviously, is constructing correspondences. Correspondences could be in totally different forms, e.g., point correspondences, line correspondences, curve correspondences, even region correspondences. Sometimes, we can easily get some geometrical primitives from images, but sometimes not. According to my understanding, there are two major methodologies: “dense” approach, and “sparse” or “feature-based” approach. The dense approach tries to build correspondences pixel by pixel, while feature-based approach tries to associate different image features. These two ideas result in totally different taste of motion and structure analysis. In this lecture, let’s get some feelings about the “dense” approach. Optical flow estimation is computationally demanding. At present there are several groups of methods for its calculation.

The optical flow determination is solved by the calculation of partial derivatives of the image signal. There are two most used methods, namely:

- Lucas-Kanade
- Horn-Schunck

One of the more popular methods for optical flow computation is Lucas and Kanade’s local differential technique. This method involves solving for the optical flow vector by assuming that the vector will be similar to a small neighborhood surrounding the pixel. It uses a weighted least squares method to approximate the optical flow at pixel (x, y). This technique has numerous advantages. Firstly, the support for the flow vector is local rather than global like the iterative technique of Horn and Schunk. This means that a good estimate without having to rely on the entire image. For some images with large homogenous regions, a global method may produce satisfactory results but for most cases, the flow vectors of different regions should not impact separate regions. Iterative techniques such as Horn and Schunk allow

vector information to spread out over the image to possibly different regions. Reinforcement of the constraint equation can serve to mitigate this, but the problem remains. Imagine two occluding objects passing, both with similar spatial gradients but with different orthogonal components.

The iterative scheme will merge these two flow field regions at the boundary and will not preserve the sharp discontinuity. The vectors produced by local techniques will not suffer this problem. The downside is that the homogeneous regions will not be filled in. The task of deciding and filling in homogenous regions is obviously important but should be accomplished at a later stage in the process, using the initial local estimates as input. In this way, the user has more control over the final optical flow field and will probably produce better results.

Direct method means that we can replace the optical flow vectors with their estimates in terms of spatial-temporal image intensity gradient.

III. OPTICAL FLOW COMPUTATION

Optical flow computation is based on two assumptions:

1. The observed brightness of any object point is constant over time.
2. Nearby points in the image plane move in a similar manner (the velocity smoothness constraint). Suppose we have a continuous image; $f\{x,y,t\}$ refers to the gray-level of (x,y) at time t .

Representing a dynamic image as a function of position and time permits it to be expressed.

As refer to (1):

$$f(x + dx, y + dy, t + dt) = f(x,y,t) + f_x dx + f_y dy + f_t dt + O\{d^2\} \tag{1}$$

where f_x, f_y, f_t denote the partial derivatives of f . We can assume that the immediate neighborhood of (x,y) is translated some small distance $(d:r,dy)$ during the interval dt ; that is, we can find dx, dy, dt as refer to (2) :

$$f(x + dx, y + dy, t + dt) = f\{x, y, t\} \tag{2}$$

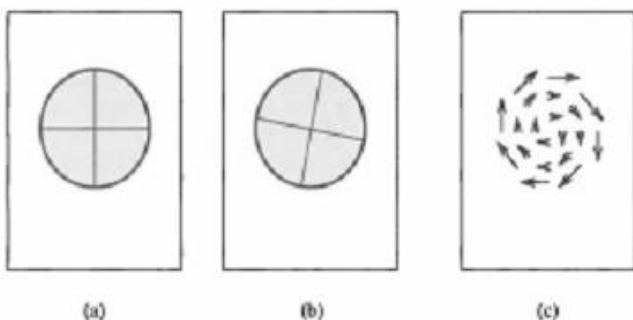
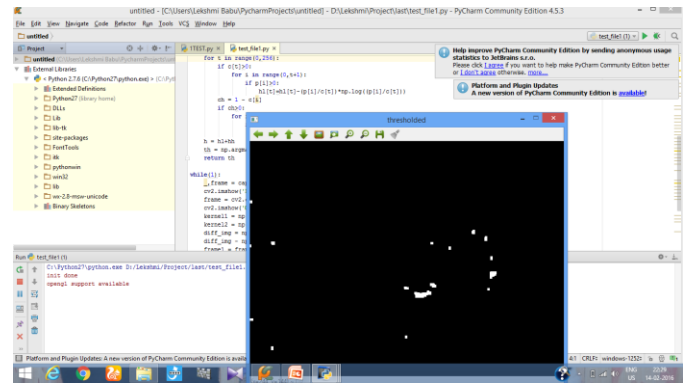


Fig. 4. Optical flow, (a) Time t1, (b) Time t2 (c) Optical flow

IV. METHODS AND MATERIALS



A. Functionalities

This software has capability to detect motion detection based on optical flow and gives certain level of motion detection which can be used as a threshold. Analyzing motion level and comparing it with predefined threshold allows raising alarm, when detected motion level is greater then the level which is considered to be safe. In addition to motion level detection, there are four types of motion detectors and all of them support highlighting of detected motion regions (can be turned on/off).

B. Types of Detection used in the Software

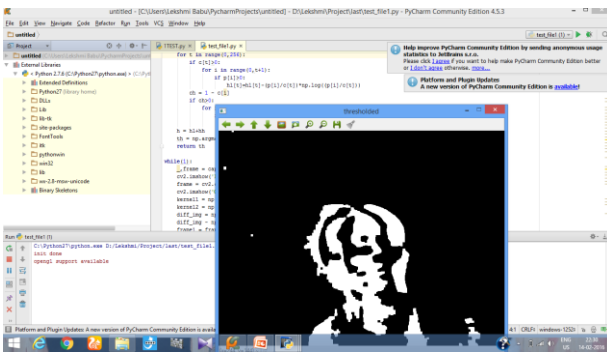
1) Two frames difference motion detector

This type of motion detector is the simplest one and the quickest one. The idea of this detector is based on finding amount of difference in two consequent frames of video stream. The greater is difference, the greater is motion level. As it can be seen from the picture below, it does not suite very well those tasks, where it is required to precisely highlight moving object. However it has recommended itself very well for those tasks, which just require motion detection.



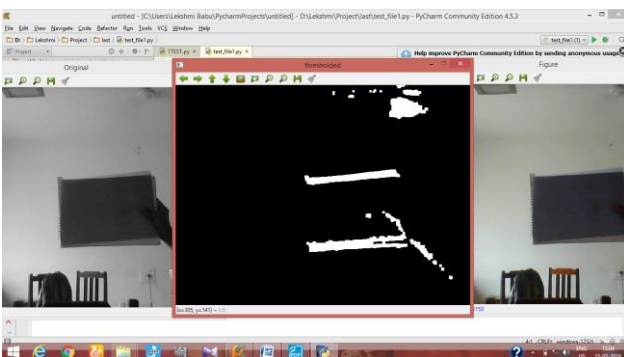
2) Motion detectors based on background modeling

In contrast to the above motion detector, these motion detectors are based on finding difference between current video frame and a frame representing background. These motion detectors try to use simple techniques of modeling scene's background and updating it through time to get into account scene's changes. The background modeling feature of these motion detectors gives the ability of more precise highlighting of motion regions. Below are demonstrated outputs of two versions of motion detectors based on background modeling. One does more precise highlight of moving objects' borders, but consumes more computational resources. Another one does less precise objects' highlight in the cost of requiring much less computational resources.



3) Counting Motion Detectors

The counting motion detector is based on the same idea of background modeling as the above motion detectors. However after that it does additional processing and different object's highlighting. Once motion regions are identified, this detector uses blob counting algorithm to find rectangles of each detected moving object. This gives the ability to report about amount of detected objects, as well as position and size of each detected object. The sizes can later being used in human recognition.



V. RESULT

According to the figures given above we found that “two frames difference” method is suitable for simple motion detection only it can not highlight specific region of moving objects. From the two types of background modeling we observed that the high precision method is good for detecting whole moving object precisely but it requires complex computational method. On the other hand Low precision method requires less calculation but it gives less precise result.

We also found that the Counting motion method is good to count the object and get their specific size.

VI. DISCUSSION

Optical flow reflects the image changes due to motion during a time interval dt which must be short enough to guarantee small inter-frame motion changes. The optical flow field is the velocity field that represents the three-dimensional motion of object points across a two-dimensional image. Optical flow computation is based on two assumptions: 1) The observed brightness of any object point is constant over time. 2) Nearby points in the image plane move in a similar manner (the velocity smoothness constraint).

VII. CONCLUSION

The proposed method does not require user interaction or parameter tuning as most of the preceding works did. The experimental results show that the proposed schemes can detect moving objects with high accuracy and robustness. While there are consecutive serried moving objects coming through the scene, they will exert negative interactions on each other. Our future work will focus on how to deal with the interactions between the consecutive moving objects. Another extension is to use the proposed approach for some consumer video applications. Optical flow computation will be in error if the constant brightness and velocity smoothness assumptions are violated. In real imagery, their violation is quite common. Typically, the optical flow changes dramatically in highly textured regions, around moving boundaries, at depth discontinuities, etc. Resulting errors propagate across the entire optical flow solution.

REFERENCES

- [1] Gibson. 1.1.. The Perception of the Visual World (Riverside Press, Cambridge. 1950)
- [2] Gibson.1.1.1.The Senses Considered as Perceptual Systems (Houghton-Mi Win, Boston, MA, 1966).
- [3] Gibson, J.J, On the analysis of change in theoptic array. Scandinavian I. Psychol. 18 (1977)161-163.
- [4] Nakayama. K. and laomis. J.M.. Optical velocity patterns. Velocity- sensitive neurons and space perception. Perception 3 (1974) 63-80.
- [5] Thompson et al., 1985; Kearney et al., 1987; Aggarwal and Martin, 1988. Thompson et al., 1984; Mutch and Thompson, 1984.
- [6] B.H. Sonka, Image Processing, Analysis, and Machine Vision [1999], pp. 757-759,808.
- [7] S. Sun, Y. Wang, F. Huang, and H. Liao, “Moving foreground object detection:viarobustSIFTtrajectories,”J.Vis.Commun.ImageRepresent., vol. 24, pp. 232–243, Apr. 2008.
- [8] Y. Liu and G. Yan, “An adaptive detection model of moving dim tar- gets based on energy difference between frames,” in Int. Conf. Space Inform. Technol., 2009.
- [9] C.Zhan,X.Duan,S.Xu,Z.Song,andM.Luo,“Animprovedmoving object detection algorithm based on frame difference and edge detec- tion,”in4thIEEEInt.Conf.ImageandGraph.,Aug.2007,pp.519–523.
- [9] S. Yoshinaga, A. Shimada, H. Nagahara, and R. Taniguchi, “Back- ground model based on intensity change similarity among pixels,” in the 19th Japan-Korea Joint Workshop on Frontiers of CV, Jan. 2013, pp. 276–280.

- [10] A. Park and H. Byun, "Object-wise multilayer background ordering for public area surveillance," in IEEE Int. Conf. AVSS, Sep. 2009, pp. 484–489.
- [11] R. H. Evangelio, M. Pätzold, and T. Sikora, "Splitting gaussians in mixture models," in Proc. 9th IEEE Int. Conf. Advanced Video and Signal-Based Surveillance, Sep. 2012, pp. 300–305.
- [12] Y. Tsaig and A. Averbuch, "A region-based MRF model for unsupervised segmentation of moving objects in image sequences," in CVPR, 2001, pp. 889–896.
- [13] F. Porikli and O. Tuzel, "Bayesian background modeling for foreground detection," in Proc. ACM VSSN, Nov. 2005, pp. 55–58.
- [14] Y. Nonaka, A. Shimada, H. Nagahara, and R. Taniguchi, "Evaluation report of integrated background modeling based on spatio-temporal features," in Proc. IEEE Comput. Soc. Conf. CVPR workshops, Jun. 2012, pp. 9–14.
- [15] S. Zhang, H. Yao, and S. Liu, "Dynamic background modeling and subtraction using spatio-temporal local binary patterns," in IEEE Int. Conf. ICIP, Oct. 2008, pp. 1556–1559.
- [16] J. L. Barron, D. J. Fleet, and S. S. Beauchemin, "Systems and experiment performance of optical flow techniques," Int. J. Comput. Vis., vol. 12, 1, pp. 43–77, Feb. 1994.
- [17] J. Xiao, H. Cheng, H. Sawhney, C. Rao, and M. Isardi, "Bilateral filtering-based optical flow estimation with occlusion detection," ECCV, 2006, pp. 211–224