FEATURE BASED OPTICAL FLOW MODEL FOR OBSTACLE DETECTION ON A CAMERA PHONE

Abdulmalik Danlami Mohammed
Department of Computer Science
Federal University of Technology, Minna
Niger State, Nigeria,
drmalik@futminna.edu.ng
https://orcid.org/0000-0002-0217-7411

Tim Morris
School of Computer Science
University of Manchester
Manchester, United Kingdom
tim.morris@manchester.ac.uk

***Abstract*----- A wide variety of assistive tools has failed to address some or all of the challenges faced by people with visual disability. For instance, obstacle avoidance poses a big challenge to visually impaired persons especially during the navigation of unknown places. The high resolution and low cost camera phone provides an alternative to record video and capture high quality images, which simplifies the computation of optical flow vectors for obstacle detection. This study, explore the benefit of camera phone in the capturing and processing of video frame into flow vectors for obstacle detection.**

***Index Terms*----Image Analysis, Optical Flows Vectors, Obstacle Avoidance, Smart phone, Nearest Flow**

I. INTRODUCTION

The problem of obstacle detection can be reduced to finding objects (static or dynamic) in a free space, which impede the movement of another object. Most obstacle detection techniques exploit the visual attributes of images such as colour [1][2] to divide the image plane into obstacle and free space thereby indicating the presence of an obstacle in a scene. The use of this visual color as an attribute for obstacle detection is prone to error because of the many constraints imposed on them. For example, in color based obstacle detection, it is assumed that, the object and background colors are different. However, in a complex background where both objects and background have similar coloration; so many false positive objects are likely to be detected as obstacles and thus degrade the accuracy of the system. Aside from the use of colours and edges for detection of obstacles in a given sequence of images, motion cues such as the optical flow have been widely used in obstacle detection in recent times[3][4][5][6][7].

Optical flow describes the spatial and temporal displacement of pixel intensity in image sequences. It has practical application in many vision based applications including 3D reconstruction, focus of expansion estimation etc. Most obstacle detection techniques based on optical flow vectors are implemented using sophisticated and expensive sensory devices like the sonar, stereo camera, and range finder[8][9][10][11]. The high resolution and low cost camera phone provides an alternative to record video and capture high quality images, which simplifies the computation of optical flow vectors for obstacle detection.

In recent time, there has been an unprecedented upsurge of camera phone in the mobile phone market. This increase is due to the significant advance made in terms of hardware and software implementation of the devices to meet the

978-1-7281-7407-5/20/$31.00 ©2020 IEEE
DOI 10.1109/SMART-TECH49988.2020.00024

current demands and challenges. For example, most of the smart phones come with high resolution and low cost cameras that are capable of capturing high quality video and still images. In addition, most smart phones are less expensive and thus affordable to many groups of people. This new developments in mobile phone technology has led to a new research direction in image processing and computer vision techniques for mobile device. Hence, accurate implementation of an obstacle detection method on a mobile phone will not only increase the independent of visually impaired persons, but also boost their confidence when navigating unknown places.

In this work, we implement an obstacle detection method on a lightweight and low cost camera phone using an image feature (e.g corner) and a matching approach to estimate the optical flow vector.

II. RELATED WORK

There are wide ranges of obstacle detection methods propose in the literature. However, not many of these methods have been implemented on a camera phone. Several obstacle detection methods make use of object attributes such as color, edge, and texture to implement the task of obstacle detection. For example, Parag et.al [1] proposed an obstacle detection algorithm using both color based segmentation of the visual scene and color homograph. While the color segmentation allows for the classification of image region as obstacles or free space after training, the color homograph on the other hand, gives an indication of a projected image feature above the ground level thereby signaling the presence of an obstacle in the visual scene. While this approach works in a simple and constrained environment, it has high false positive rate detection because of the similarity in appearance between objects and the ground. Another approach that exploit color feature of an image for detection of obstacle is presented in [2]. In this approach, each image pixel is considered an obstacle or a free space based on its color information. To be able to understand and adapt to more features of the free space, the system is trained through learning of more ground features of the environment. This training process is done manually by driving a mobile robot through the visual scene/environment. The approach is robust to lightening condition since the many ground appearances is learned under different lightening conditions, it is however, vulnerable to failure because of similar coloration of object and background. The work by [12] proposed an obstacle detection system based on color blob tracking technique in a sequence of images. In their approach, a color segmentation of the image is performed, followed by the grouping of similar region that share common color properties to form color blob. The final step of the algorithm is to find corresponding colour blob in the next image frame using the centroid property of the color blob couple with its area and aspect ratio. While this approach boasts of good performance in a wider environment, it is prone to error because of variation in aspect ratio and image area thus degrading the accuracy of tracking and by extension the detection of obstacle.

In recent time, a powerful and important motion cue, such as optical flow has been widely used for obstacle detection in the path of moving vehicle or mobile robots. For more discussion on optical flow field estimation, please refer to section 3 of this text. Several techniques for the estimation of optical flow field from sequence of images have been reported in the literature. For example, to detect a stationary object in the path of moving vehicle with a mounted camera, the work presented in [14], evaluated the difference between the calculated optical flow field and the estimated optical flow field of the scene assuming a translational motion of the camera relative to the scene. Furthermore, the approach exploited the orientation of the optical flow field to detect the motion of other objects in the visual scene. A similar approach that employed an optical flow for obstacle detection is presented in [15]. In the estimation of the optical flow field, Toby et.al [15] uses Harris corner detector to find salient features such as corners in the image frame, following which a correlation matching techniques is implored to find the corresponding flow field in the next image frame. The optical flow field in this case provides range information to object that is crucial for obstacle avoidance and hence, enhance navigation of mobile robots. In [16], an obstacle detection system based on optical flow is proposed. The proposed method detect discrete obstacle by evaluating the difference between the calculated flow line and a reference flow line that represents the expected slope of a terrain where the vehicle is expected to traverse. Kai-Tai et.al [17] exploits the intensity gradient of an image and matching techniques to compute optical flow field. This method is implemented on a guide robot to avoid obstacle in the path of travelling person with partial or complete blindness. In contrast to [17], our work exploits the key points (e.g corner) in an image and matching techniques to find the corresponding key points in the following image frame on a camera phone.

III. METHODOLOGY

The block diagram of figure 1 shows the workflow of our obstacle detection method. The obstacle detection approach is based on estimating the difference between a template optical flow vector and a query flow vector,

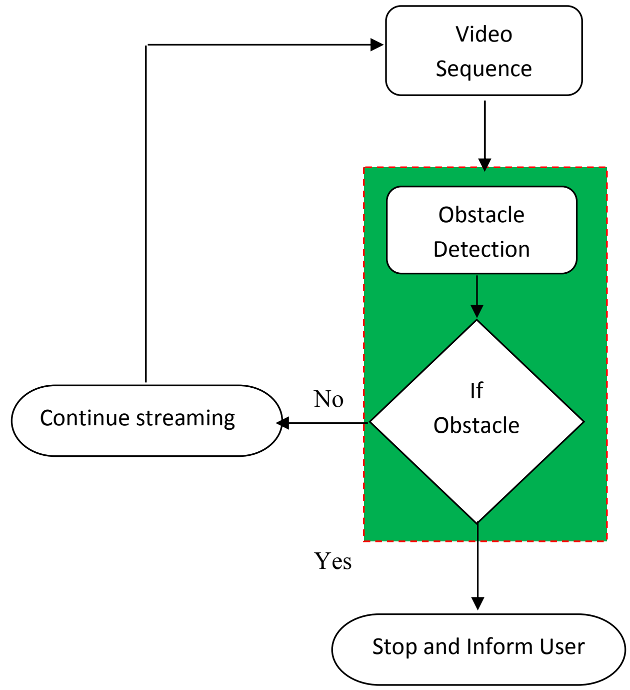


Fig 1. Block diagram of Obstacle detection

As shown in figure 1, the system begins with the capturing of video sequences and for each frame in the sequence an optical flow vector is computed. The difference between the computed flow vectors and the reference flow vectors is calculated and the result of which determines the presence or the absence of an obstacle in the path of mobile user. Mobile users are informed if an obstacle is detected, otherwise video sequence continues.

In this paper, we assumed a simple translational motion of the camera as held by the user who is walking in a forward direction such that the camera focus is approximately parallel to the ground plane. The obstacle detection method proposed in this study consists of three steps.

- The first step is concerned with the estimation of optical flow vectors. Here, the Nearest-Flow method is employed for the computation of optical flow vectors for individual frames.

- The second step is in the estimation of reference flow vectors, which are obtained through the approximation of the focus of expansion of the flow field and the time before collision occurs. The components of the reference flow vectors are computed in a similar way to the one proposed by [14].
- In the final step, the difference between the computed optical flow vectors and reference flow vectors is evaluated. The result of this evaluation is compared with a certain threshold to determine the presence of an obstacle in the path of an individual.

The computation of optical flow vectors has been thoroughly discussed in [13]. Hence, in this paper, we restrict the discussion of our proposed obstacle detection on the second and third steps.

A. Reference flow vectors

Assuming a translational motion by the camera, the components of the reference flow vector $V_r(u, v)$ is estimated using the following equation:

$$u = \frac{(x - p_x) - (FOE_x - p_x)}{\tau} \quad (1)$$

$$v = \frac{(x - p_y) - (FOE_y - p_y)}{\tau} \quad (2)$$

Where:

FOE_x, and FOE_y denote the focus of expansion of the flow field in the x and y directions respectively, τ represents the time before contact; p_x and p_y are projection points of the optical axis.

The focus of expansion of the flow vectors can be estimated as follows:

$$FOE = (A^T A)^{-1} A^T b \quad (3)$$

Where:

A denotes the vector of all spatial gradients; b is the vector of all temporal gradient.

Thus:

$$A = \begin{matrix} a_{00} & a_{01} \\ \cdots & \cdots \\ a_{n0} & a_{n1} \end{matrix}, b = \begin{matrix} b_0 \\ \ldots \\ b_n \end{matrix} \quad (4)$$

$$A = \begin{bmatrix} a_{00} & a_{01} \\ \vdots & \vdots \\ a_{n0} & a_{n1} \end{bmatrix}, \quad b = \begin{bmatrix} b_0 \\ \vdots \\ b_n \end{bmatrix}$$

Each element in the matrix above corresponds to a component of the flow vectors. i.e $a_{i0} = v$; and $a_{i1} = u$, $b_i = xv - yu$

The time to contact before collision is estimated as follows:

$$\tau = \frac{z}{V} \quad (5)$$

Where:

z denotes the distance from a pixel to the FOE, while V is the constant velocity that corresponds to the length of optical flow vectors., $(\frac{\partial y}{\partial t})$.

With all these motion parameters in place, we can now compute the component of the reference flow vectors (1 and 2).

B. Evaluation of different optical flow vectors

Following the estimation of the reference vector, it is time to evaluate the difference between the calculated optical flows vectors and the reference flow vector:

$$V_D = V_c - V_r \quad (6)$$

Where:

V_D denotes the difference between V_c calculated optical flow and V_r reference flow estimated using FOE and time to contact. To avoid division by zero and obtained an reliable result in instance where the length of the reference vector is less than the calculated optical flow, we used an approach similar to the one proposed in[14] to find the ratio of the absolute values of the difference vector $|V_D|$ and the reference vector $|V_r|$. Subsequently, this ration is compare against a threshold θ . However, in contrast to[14] where the threshold value referred to as dimensionless and set to a constant value of 0.3, we obtained our threshold value by first computing the probability density function for the extracted reference vectors obtained from image sequence when no obstacle is in view and flow vectors with a obstacle. Using the expression in equation 7, we found our threshold to be equal to 0.4.

$$\theta = \frac{pdfV_r}{pdfV_r + pdfU} \qquad (7)$$

Where:

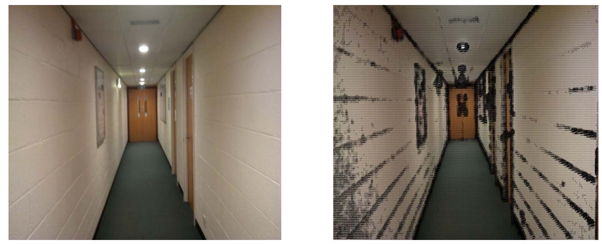
θ represents the threshold value that is compare against a ratio of an absolute values of the difference flow vectors and reference vectors. The result of which determine the presence or absence of an obstacle. While $pdfV_r$ denotes the probability density function for reference flow vector, $pdfU$ denotes the probability density function for flow vectors with anticipated obstacles.

Consequently, an obstacle is detected:

$$if\ V_c > V_r \quad \text{and} \quad \frac{V_D}{V_r} \begin{cases} > \theta & = obstacle \\ < \theta & = no\ obstacle \end{cases} \qquad (8)$$

IV. RESULT AND DISCUSSION

In this paper, experiment on our obstacle detection method on a mobile phone was conducted by walking around the corridors within a building. While conducting this experiment, the system was configured to process 30 frames per second and all the generated data (e.g flow vectors in the x and y coordinate and the angular error) were recorded and saved in the system log file as the system executes. This will help in further analysis. For instance, knowing the number of image location correctly identified as obstacles will provide us with the right information to judge the potential of using the system in environments other than the tested one. A sample result of computing optical flow is shown in figure 2. As can be seen from figure 2, optical flows are calculated along the walls and to the far end door of the corridor. This shows that both the door and wall most be avoided in order to have smooth navigation along the corridor. In Figure 3, the position of the flow vectors computed for two successive frames is shown.



(a) (b)

Fig 2. (a) Corridor of a building captured with camera phone (b) The optical flow computed along the corridor of building in (a).

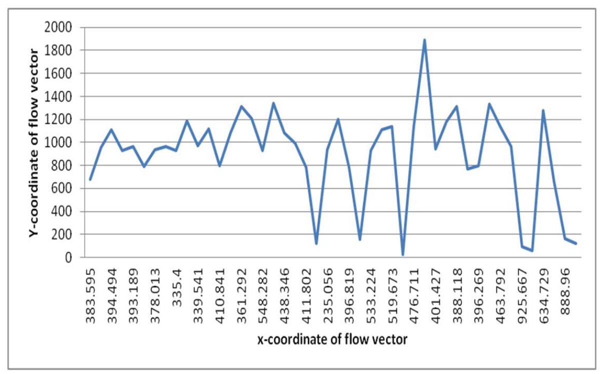


Fig 3. Position of flow vectors computed for two frames

V. CONCLUSION AND FUTURE WORK

Even though, our system is designed to detect obstacle in the path of mobile user by evaluating the difference of flow vectors between successive frames in a video sequence, it is worth mentioning here that the time–to-contact information is another property that could be used for obstacle detection given the exact optical measurements. In order to achieve an accurate estimation of time-to-contact values, other assumptions such as the focal length of the camera, image size used for subsampling, etc has to be made. In addition, most of the equations that led to the estimation of this time-to-contact have to be optimized in order to achieve the desired output. Consequently, it is our hope that future work will implore time-to-contact information approach for obstacle avoidance.

REFERENCES

[1] Parag H. Batavia and S. Singh. Obstacle detection using adaptive colour segmentation and colour stereo homography. International Conference on Robotics and Automation, vol.1 pp705-710,2001

[2] I. Ulrich and I. Nourbakhsh. Appearance-based obstacle detection with monocular colour vision., nternational Conference on Artificial Intelligence, 2000.

[3] N. Aswini and S. V. Uma. Obstacle Detection in Drones Using Computer Vision Algorithm. Advances in Signal Processing and Intelligent Recognition System pp. 104–114, 2018.

[4] C. Braillon, C. Pradalier, J. Crowley, C. Laugier. Real-time Moving Obstacle Detection Using Optical Flow Models. IEEE Symposium on Intelligent Vehicle pp.466-471. 2006. Tokyo, Japan.

[5] L. Xui, Z. Dai and J. Jia. A Scale Invariant Optical Flow. Proc ECCV pp385-399. 2012

[6] G. Farneback. Two-frame motion estimation based on polynomial expansion. Image Analysis, pp363-370. 2003

[7] B. K.P, Horn, B. G. Schunk. Determining optical flow. Techniques and Applications of Image Understanding, vol. 281, pp319-326. 1981

[8] S. Hrabar, P. Corke, G. Sukhatme, K. Usher, and J. Roberts. Combined optic-flow and stereo-based navigation of urban canyons for a UAV. 2005.

[9] A. Talukder and L. Matthies. Real-time detection of moving objects from moving vehicle using dense stereo and optical flow. October 2004

[10] K. Young-Geun and K. Hakil. Layered ground floor detection for vision based mobile robot navigation. In International Conference on Robotics and Automation, New Orleans, april 2004, pp. 13–18.

[11] M. Perrollaz, R. Labayrade, C. Royere, N. Hautiere, D. Aubert. Long Range Obstacle Detection Using Laser Scanner and Stereovision. IEEE Symposium on Intelligent Vehicles , 2005, Tokyo, Japan.

[12] B. Heisele & W. Ritter. Obstacle Detection Based on Color Blob Flow. IEEE Symposium on Intelligent Vehicles. 1995. Detroit, MI, USA

[13] M. D. Abdulmalik, T. Morris. Optical Flow Estimation Using Local Features. Proceedings of the World Congress on Engineering, 2015.Vol I WCE 2015, July 1 - 3, 2015, London, U.K

[14] W.Enkelmann, V.Gengenbach, W. Kruger, S. Rossle, W.Tolle. Obstacle detection by real-time optical flow evaluation. Imaging and Vision Computing, Vol.1 No.3 pp160-168. 1990.

[15] G. -S.Young, T. -H. Hong, M. Hermma and J.C.S. Yang. Obstacle detection for a vehicle using optical flow., Proceedings of Intelligen Vehicle Symposium, pp185-190. 1992

[16] G. -S.Young, T. -H. Hong, M. Hermma and J.C.S. Yang. Obstacle detection for a vehicle using optical flow., Proceedings of Intelligen Vehicle Symposium, pp185-190. 1992

[17] K. -T. Song. J.-H. Huang. Fast optical flow estimation and its application to real-time obstacle avoidance., International Conference on Robotics and Automation, pp2891-2896. 2001