Australian Journal of Basic and Applied Sciences, 9(26) Special 2015, Pages: 26-29

ISSN:1991-8178

Australian Journal of Basic and Applied Sciences

Journal home page: www.ajbasweb.com

Heterogeneous Optical Flow-Based Tracking for Traffic Control

Fatimah Khalid and Zainal Abdul Kahar

Universiti Putra Malaysia, Multimedia Department, Faculty of Computer Science and Information Technology, 43400 Serdang, Selangor, Malaysia

ARTICLE INFO

Article history:
Received 13 June 2015
Accepted 5 August 2015
Available online 12 August 2015

Keywords:
Object Detection, Tracking, Optical Flow

ABSTRACT

This paper introduces a new approach of heterogeneous optical flow based tracking for traffic control that manages traffic efficiently by the adaptive of optical flow tracking. As we know, existing traffic light system is controlled by a time relay and it is configured by priority and probability of traffic occurrence. However, this system incapable to manage traffic in real-time condition that cause unnecessary waiting time and inefficient traffic control that leads to heavy congestion as well as causing bad air pollution. Common time relay unintentionally allows traffic light to remain green on empty road. This awareness leads to this approach that will remarkably clear traffic and fully operated by the optical flow congestion detection and disregard of gaussian noise and luminance and chrominance differences (e.g night and daylight). This approach consist fallback plan which will switch to time relay whenever technical failures occur without affecting the ongoing traffic flows. Technical failure is triggered by certain threshold of error detection. The contribution of this approach is the adaptability of optical flow tracking over luminosity changes and poor contrast.

To Cite This Article: Fatimah Khalid and Zainal Abdul Kahar., Heterogeneous Optical Flow-Based Tracking For Traffic Control. ***Aust. J. Basic & Appl. Sci.****, 9(26): 26-29, 2015*

INTRODUCTION

The past decades, traffic congestion has significantly become our notable concern in a society due to the increase of vehicles. Solutions include widening roads and manually using police traffic and road diversion. Various studies have been conducted to improve the intelligent traffic system (ITS), which homogenised electronic, network, sensing and control. To solve this issue, we would like to propose heterogeneous approach using optical flow and cumulative distribution function technique for illumination improvement. The main objectives of this approach is to optimise the consumption of cpu processing for real-time system (RTS) using lower-end devices. RTS normally requires consistent image processing.

RTS in traffic flow measurement involves real-time image acquisition, image enhancement, and vehicle flow detection using object tracking (Bedi and Khandelwal, 2013).

Our approach using optical flow tracking can be an important method for traffic analysis similar to object tracking which can track and detect optical fields, identify moving objects, and calculate the velocity of objects from the origin to destination (Indu *et al*., 2011). However, this approach requires further improvement such as image enhancement technique, and fastest feature extraction from the acquired images. In this paper, we will explain in detail about feature selection based on pre-determined points for traffic system where the extraction points are selected in early stage by estimation rather than the common edges extraction. The scope of this proposed idea is within optical flow tracking and feature extraction.

Related Work:

In this paper, our main focus is in motion tracking using optical flow technique for lower-end devices that can be embedded into existing traffic system. Therefore, we will describe 1) feature extraction in visual scene and 2) approaches using optical flow in motion tracking.

A. Invariant Feature Detection:

In image processing, we have to identify variables that varying the image properties (Ma *et al*., 2003) : angle of view, illumination, and distortion. Among these image variations, the perspective difference between each frame constitutes a significant factor, especially when the camera baseline is large between the views. In these wide baseline cases, the feature matching problem

Corresponding Author: Fatimah Khalid, Universiti Putra Malaysia, Multimedia Department, Faculty of Computer Science and Information Technology, 43400 Serdang. Selangor. Malaysia.
Tel: +603 89471717

requires the features to maintain invariant properties for large differences in viewing angles and camera translation. The features also need to be discriminative in order to be used for recognising the scene repeatedly and robustly. In this sense, these invariant features are often also referred to as landmark features. The scale invariant feature transform (SIFT) (Lowe, 2004). is an example to extract the key points from a captured video frame. SIFT is a technique to get the interest points (key-points) in the image and calculate a descriptor around that point which can be used for matching. The SIFT key-points is process by convolving the image with Gaussian filter varies by scales and then retrieved the Difference of Gaussian (DoG) throughout the images. DOG in scale space are detected as key points, characterised accurately by four components: the pixel location (x, y) the scale α, and the major orientation θ. The dimension of the descriptor depends on the number of histograms and the number of bins in each histogram, i.e., a 4 × 4 array of eight bin histograms results in 4 × 4 × 8 = 128 dimensions. The key points selected by the SIFT algorithm are shown to be invariant to scale and orientation changes, and the descriptors remain robust to illumination changes. The challenge of using SIFT extraction is the consumption of memory which lead us to use pre-determined feature extraction (Lee and Höllerer, 2009). Another example that close to our proposal is the random seeding where pixels can be selected randomly. There are three information is required: image width, height, and number of pixels to be extracted from the scene.

B. *Optical flow based tracking:*

Tracking features between two consecutive image frames is considered a small-baseline tracking problem in the sense that the transformation from the image at time t to the image at time t + dt for small dt can be created using the translational model. Computing the optical flow of the feature points involves frame-to-frame feature tracking (Lucas and Kanade, 1981) (Lee and Höllerer, 2009). Lucas - Kanade optical flow is a block object motion detection (Tekalp, 1995). where the approach classify object's movement into four directions such as left, right, up, and down. As a result, the region of moving objects can be extracted and the corresponding region is then labeled based on motion direction. After detecting moving objects from background, a set of feature points inside the object were extracted and predicted the corresponding feature points in the next frame. If over 60% of feature points are restored, the set of feature points are too small and not proper for tracking, therefore, a redefine new set of points must be extracted Shina, J., Kima, S., Kangb, *et al.*, 2005). The algorithm tracks a set of feature points based on optical flow. A missing feature point during the tracking is restored by using both temporal and spatial information inside the predicted region. One important contribution of this work is to provide a restoration process for missing feature points, which occurs at almost every frame under realistic, noisy environment.

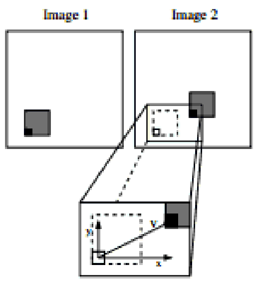

Fig. 2.0: Lucas Kanade's approach.

Methodology:

Optical flow tracking involves reading of motion velocity that associated with image translation frame by frame. Most approaches of optical flow derive from illumination different between two images or DOG. In Lucas and Kanade's method, a small region of image corresponds as an object for optical flow tracking as illustrated in figure 2.0. where the extracted feature points predicted. The image is divided as N x N size. In optical flow each pixel is coordinated as vector (x, y). U and V are the displacement of x and y with the notation of d = (d_x, d_y). For I_x and I_y and It is the image brightness for horizontal, vertical coordinate and time (Aires, *et al.*, 2008).

A. *Pre-determined feature extraction:*

In our propose solution for optical flow tracking, we simply used pre-determined feature extraction where DOG is not applicable for traffic motion tracking. The reason is when the road is empty, DOG will extract only edges with different gaussian, in this case the most important feature: the road will not be extracted as key point. However, DOG may be used to extract feature during the motion detection, however it will cost memory consumption. As a result we used pre-determined feature extraction based on assumption of visual scene. There are N x N size of key points that can be reduced by applying the division of subframe as illustrated below.

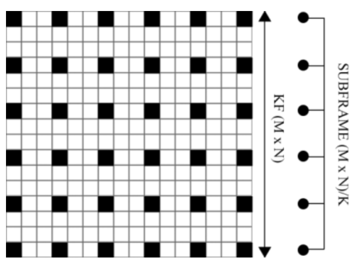

Fig 3.1a: pixel selection for Optical flow tracking.

As illustrated in figure 3.1, 5 pixels are extracted from the block region of W=width and H=height. The colour feature for each pixel with coordinate (x, y) is extracted. Each key points represent in magnitude 1 over N x N size.

$$\alpha(x) = \left(\sum_{i=1}^{n} i\right).\,\delta$$

$$\beta(y) = \left(\sum_{j=1}^{n} j\right).\delta$$

The extracted key points location is represented as (α,β) notation where the size is clamped between 0 to 1. The information of extracted feature is used to track the motion f (α,β).

B. Motion tracking:

As illustrated in figure 3.1b, the extracted features is used to process the motion of optical over the visual frames. At first, a sample image with size of (M x N) is acquired. The sample image is defined as KF, and the subframe of KF is denoted by constant K. The location of each pixel is registered as the origin location. Next, the RGB value is retrieved. If the pixel intensity changed, intensity function is called and neighbouring node is check within certain displacement threshold. The intensity function is represented as I (x_t, y_t) where t = time with the i = intensity threshold.

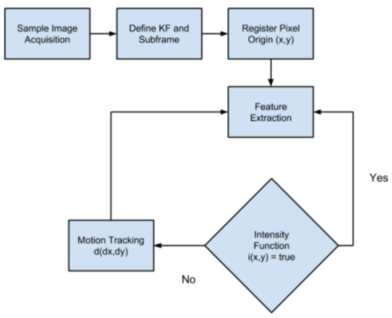

Fig. 3.1b: Motion Tracking.

When function of $i(x_t, y_t) \neq f(\alpha,\beta)$, means the optical flow displacement occurred. In this case, the d(dx, dy) is searched from the origin as middle of tracing with s=scale and d=direction in degree as illustrated in figure 3.1c.

In figure 3.1c, the scale of tracking is the range of optical movement from pixel to pixel. The higher the scale, the fastest the tracking but with low accuracy. In our experiment, we assumed that the suitable scale for object detection with is 50 pixel range.

C. Traffic control system:

The incorporation of feature extraction and motion tracking, we're able to produce heterogeneous traffic system based on visual detection. However, there are confounding variables such as gaussian noise and distortion that need to be filtered by using the formula below:

$$dist[\,c(x,y)\ c(x',y')\,] \;>\; k$$

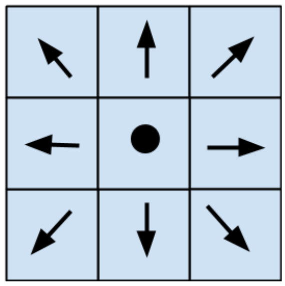

Fig. 3.1c: Optical flow tracing with S=1.

This formula indicate the distance between key point at location x,y and x',y' with the threshold of k. The threshold will allow slight movement of pixel if its distorted by unnecessary effect from the environment such as wind or gaussian noise from a camera.

D. Image Acquisition:

Sample images acquisition is taken from low-end device that has the ability to communicate through ad-hoc network or simple bluetooth pairing.

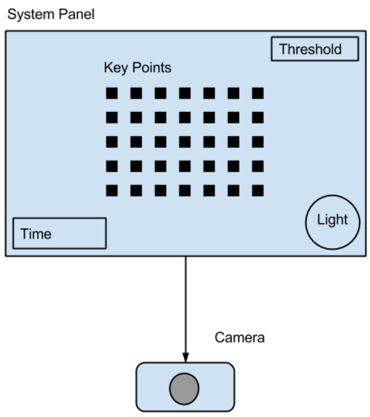

Fig. 3.1d: ITS Visual processing.

As illustrated in figure 3.1d is the blueprint for ITS visual processing (ITSvp) device that become part of the control in the traffic system. This device will be integrated with existing legacy traffic that can receive input.

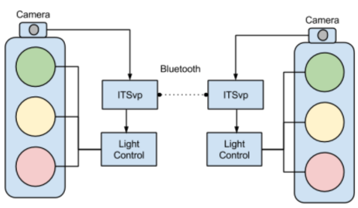

Fig. 3.1d: ITS System.

The ITS System as illustrated above consists two main components: The visual processing and the light control. The communication between these two traffic lights is by using wireless connection that distributed the time mapping.

Experiment:

As illustrated below, the system is tested using 3D simulation for stage alpha. There are six key points selected for feature extraction. Each point will demonstrate of motion tracking which control the traffic light whenever the car is detected.

Based on figure 5.0a, there are six key points being selected for feature extraction. These key points track optical flow and analyse if there is a vehicle waiting for green light. The ITSvp will change it state to green light when vehicle detected or remain red light if no vehicle detected.

Table I: Motion Tracking and Light Management.

Motion	Traffic Light A	Traffic Light B
No	Red	Green
Yes	Red	Green
Yes	Green	Red
No	Red	Green
No	Red	Green
No	Red	Green
Yes	Red	Green
Yes	Green	Red

According to the table above, by using ITSvp, it gives the priority to Traffic Light B as the main road and significantly reduce the traffic congestion at Traffic Light B where most of the green light occurred.

Fig. 5.0a: Traffic Simulation.

Conclusion:

By having this approach, it is possible for the ITSvp to become real-time and saving the operational cost. Compared to different type of feature extraction, this will support the tracking of incoming traffic, rather than by analysing the traffic itself and labelling each object that will cause CPU consumption and error prone due to the capability of the device itself. For the future plan we would like to make the ITS system distributing message through an ad-hoc network, and capable to handle all kind of traffic.

REFERENCES

Aires, K R.T., M. Andre, A.M. Santana, A.A.D. Medeiros, 2008. Optical Flow Using Color Information: Preliminary Results. SAC, 16-20.

Bedi, S.S. and R. Khandelwal, 2013. Various Image Enhancement Techniques- A Critical Review. International Journal of Advanced Research in Computer and Communication Engineering, 3(3):1605-1609.

Indu, S., M. Gupta and A. Bhattacharyya, 2011. Vehicle Tracking and Speed Estimation using Optical Flow Method. International Journal of Engineering Science and Technology, 3(1): 429-434.

Lee, T. and T. Höllerer, 2009. Multithreaded Hybrid Feature Tracking for Marker less Augmented Reality. IEEE transactions on visualization and computer graphics, 15(3): 355-368.

Lowe, D.G., 2004. Distinctive Image Features from Scale-Invariant Keypoints. Int’l J. Computer Vision, 60(2): 91-110.

Lucas, B. and T. Kanade, 1981. An Iterative Image Registration Technique with an Application to Stereo Vision. Proc. Seventh Int’l Joint Conf. Artificial Intelligence (IJCAI ’81), 674-679.

Ma, Y., S. Soatto, J. Kosecka and S. Sastry, 2003. An Invitation to 3D Vision, from Images to Models. Springer Verlag.

Shina, J., S. Kima, S. Kangb, S.W. Leec, J. Paika, B. Abidid and M. Abidid, 2005. Optical flow-based real-time object tracking using non-prior training active feature model. Science Direct, 204-218.

Tekalp, A.M., 1995. Digital video processing. Englewood Cliffs, NJ: Prentice-Hall.