Available online www.ejaet.com

European Journal of Advances in Engineering and Technology, 2015, 2(2): 34-39

Research Article ISSN: 2394 - 658X

Multiple Object Tracking using Kalman Filter and Optical Flow

Sanjivani Shantaiya[1], Kesari Verma [2] and Kamal Mehta[3]

[1]*Dept. of CSE, Disha Institute of Management and Technology (*DIMAT*), Raipur, India*
[2]*Dept. of MCA, National Institute of Technology Raipur, India*
[3]*Dept. of CSE, NIRMA University, Ahmadabad, India*
sanjivaninice@gmail.com

ABSTRACT

In this paper we presented a tracking of multiple objects from a given video dataset. Multiple objects can be tracked simultaneously using Kalman filter and optical flow algorithm. We presented improved optical flow algorithm which not only gives better accuracy but also handles occlusion in a video. So, improved optical flow algorithm is found to be more promising as it gives better accuracy in less computation.

Key words: Object tracking, Kalman filter, Optical flow, Occlusion

INTRODUCTION

Object tracking is a crucial task within the field of computer vision. There are three important steps in video analysis:-interesting moving objects detection, tracking of such objects from frame to frame, and analysis of object tracks to recognize their behaviours. The complexity of object tracking is due to the noises in images, scene illumination changes, complex object motion, and partial and fully objects occlusion. Most of the tracking algorithms assume that the moving object moves in smooth and no sudden change.

RELATED WORK

Numerous techniques have been proposed for multiple object tracking. However, in this section only few well-known techniques have been described in two different aspects:
1) Kalman filtering for objects tracking; and 2) Optical flow for objects tracking

Tracking Using Kalman Filters

Kalman filter recursively estimates the state of the target object. Kalman filtering is vastly used in different domains like object tracking, economics and navigation systems. But here we would only review it for object tracking. A new method was presented by Liu et al. [1] which combine properties of EKF and unscented Kalman filter (UKF) for non-linear object tracking. Here, EKF is kept conventional but the deterministic sample is taken by unscented transformation. Then posterior mean of nonlinearity is noted by propagating sample, but the posterior covariance of nonlinearity is kept linear. Berclaz et al. [4] propose an algorithm for frame-by-frame detection and linking the trajectories of an unknown number of targets for multi-object tracking using K-shortest path optimization. Zhai et al. [5] propose an approach to track an object by a dynamic model from a finite set of models. As the single-model assumption could cause tracker unstable if the target has complex trajectory or the camera has abrupt ego-motions.

Tracking Using Optical Flow

The Lucas & Kanade [7] algorithm presented a new method that uses spatial intensity gradient information to direct the search for the position that yields the best match. Horn and Schunck [2] assumed that reflectance varies smoothly and has no spatial discontinuities. This assures them that the image brightness is differentiable. Object tracking for moving object through motion vector is calculated through optical flow algorithm and Blob analysis for binary feature of an image is calculated. Tracking of object is measures by the position done by tracking in region filtering and the information of the object is created an estimation of new object [8]. Optical flow estimation is used in many applications. Vehicles navigation, video image reconstruction and object tracking are some examples [6]. Through optical flow estimation, motion parameters of moving objects can be obtained and at the same time, phenomena of occlusion and overlapping of objects may be avoided as far as possible [9]. Even though several researches are performed on tracking moving objects they suffer from the issue of occlusion handling.

OBJECT TRACKING USING KALMAN FILTER

Kalman filter is region based method for finding the regions of object in the next frame. The center of object is finding first, and then uses Kalman filter for predict the position of it in the next frame. A Kalman filter is used to estimate the state of a linear system where the state is assumed to be distributed by a Gaussian. Kalman filtering is composed of two steps, prediction and correction as shown in Fig. 1.For the motion model of a moving object (which contains some kind of dynamic noise), and some noisy observations about its position, the Kalman filter provides an optimal estimate of its position at each time step. The optimality is guaranteed if all noise is Gaussian. Then the filter minimizes the mean square error of the estimated parameters (e.g. position, velocity). The Kalman filter is an online process, meaning that new observations are processed as they arrive. To formulate a Kalman filter problem, we require a discrete time linear dynamic system with additive white noise that models unpredictable disturbances. The Kalman filter tries to estimate the state a $\in R^n$ of that system which is governed by the vector difference equation:

$$a_{\bar{K}} = Xa_{k-1} + Yb_k + c_{k-1} \tag{1}$$

with a measurement: $$\mathrm{t_k} = \mathrm{Ia_k} + \mathrm{d_k} \tag{2}$$

The random variables c_k, d_k represent the process and measurement noise respectively. They are assumed to be zero mean, white noise with covariance matrixes Q, R respectively. The matrix A is called the state transition matrix and relates the previous state a_{k-1} to the current state a_k, if no noise was present. The size of A is $n \times n$. Matrix B is optional and relates the control input (if any) $b_k \in R^l$ to the state a_k. Finally, the $n \times l$ matrix, relates the measurement t_k to the state a_k.

The Kalman filter maintains the following two estimates of the state:

- $\hat{a}(k|k-1)$, which is an estimate of the state at time-step k, given knowledge of the process up to step $k-1$. It is an a priori state estimate at time-step k.
- $\hat{a}(k|k)$, which is an estimate of the process state at time-step k given the measurement t_k. It is an a posteriori estimate of the state at time-step k.

It also maintains the following two error covariance matrices of the state estimate:

- $E(k|k-1)$, which is the a priori estimate error covariance of $\hat{a}\ (k|k-1)$
- $E(k|k)$, which is the a posteriori estimate error covariance of $\hat{a}(k|k)$

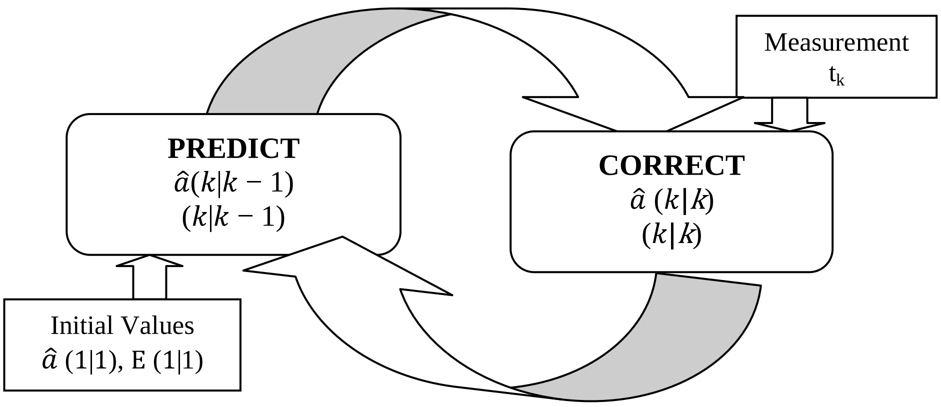

Fig. 1 The Kalman filter Predict/Correct model

A recursive minimum mean-square estimator, such as Kalman, operates in two phases on each time-step k. The firstone is the prediction of the next state estimate $\hat{a}\ (k|k-1)$ using the previous one. The second is the correction of that estimate using the measurement, to obtain $\hat{a}\ (k|k)$. Initially, $\hat{a}$ (1|1) and E(1|1) are considered known. To maintain those estimates, the following operations take place. In the prediction step:

1. State prediction: $\hat{a}\ (k|k-1) = A \cdot \hat{a}\ (k-1|k-1)$ (3)
2. Error covariance prediction: $(k|k-1) = A \cdot (k-1|k-1) \cdot A^T + Q$ (4)

In the correction step:

3. Measurement prediction: $\hat{t}\ (k|k-1) = H \cdot \hat{x}(k|k-1)$ (5)
4. Residual $r_k = t_k - \hat{t}\ (k|k-1)$ (6)
5. Measurement prediction covariance: $U_k = I \cdot (k|k-1) \cdot I^T + R$ (7)
6. Kalman gain: $C_k = (k|k-1) \cdot I^T \cdot U-1$ (8)

7. State update: $\hat{a}(k|k) = \hat{a}\ (k|k-1) + \mathrm{C}_k r_k$ (9)

8. Error covariance update: $(k|k) = (k|k-1) - \mathrm{C}_k \cdot \mathrm{U}_k \cdot \mathrm{C}_k{}^T$ (10)

So to initialize the Kalman filter, we have to define the state transition matrix A, the state measurement matrix H, the two noise covariance matrices R, Q and at each time step to feed the filter with a measurement z_k [10].

Multiple objects can be tracked easily from the video dataset as shown in Fig. 2, while kalman filter algorithm tracks multiple objects but it is failed to track occluded objects as shown in Fig 3. Also tracking is somewhat difficult if there is change in velocity of moving objects.

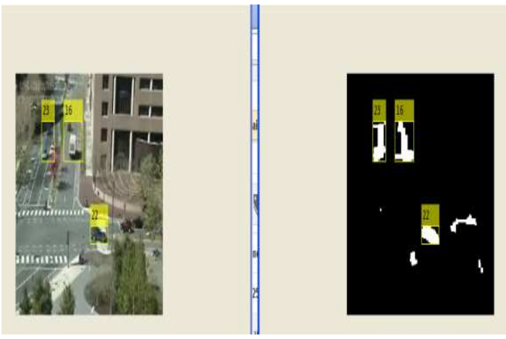

Fig. 2 Object tracking by Kalman filter

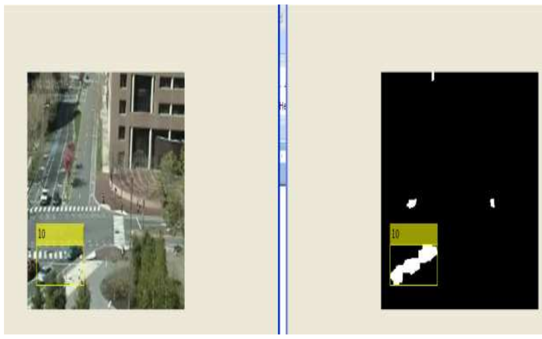

Fig. 3 Object tracking by Kalman filter under occlusion

OBJECT TRACKING USING OPTICAL FLOW

Optical flow or optic flow is the pattern of apparent motion of objects, surfaces, and edges in a visual scene caused by the relative motion between an observer (an eye or a camera) and the scene. It is the displacement field for each of the pixels in an image sequence. It is the distribution of the apparent velocities of objects in an image. By estimating optical flow between video frames, one can measure the velocities of objects in the video. In general, moving objects that are closer to the camera will display more apparent motion than distant objects that are moving at the same speed. Optical flow estimation is used in computer vision to characterize and quantify the motion of objects in a video stream, often for motion-based object detection and tracking systems.
The experimental brightness of any object point is constant over time. Close to 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.

- Assume each pixel moves but does not change intensity
- Pixel at location (x, y) in frame1 is pixel at (x+Δx, y+Δy) in frame2.
- Optic flow associates displacement vector with each pixel.

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. For making computation simpler and quicker we transfer the real world three dimensional (3-D+time) objects to a (2-D+time) case. Then we can describe the image by of the 2-D dynamic brightness function of I(x, y, t). Provided that in the neighbourhood of pixel, change of brightness intensity does not happen motion field, we can use the following expression

$$I\ (x, y, t) = I\ (x + \delta x, y + \delta y, t + \delta t) \tag{11}$$

Using Taylor series for the right hand part of the (1) we obtain

$$\mathrm{I}\ (\mathrm{x}\ + \delta\mathrm{x}, \mathrm{y}\ + \delta\mathrm{y}, \mathrm{t} + \ \delta\mathrm{t}) = \ \mathrm{I}\ (\mathrm{x}, \mathrm{y}, \mathrm{t}\ ) + \frac{\partial I}{\partial x}\delta\mathrm{x} + \frac{\partial I}{\partial y}\delta\mathrm{y} + \frac{\partial I}{\partial t}\delta\mathrm{t} + \mathrm{H.O.T} \tag{12}$$

From equation (11) and (12), with neglecting higher order terms (H.O.T.) and after modifications we get

$$I_x . v_x + I_y . v_y = -I_t \tag{13}$$

or in formal vector representation

$$\nabla I . \vec{v} == -I_t \tag{14}$$

where ∇I is so-called the spatial gradient of brightness intensity and $\vec{v}$ is the optical flow(velocity vector) of the image pixel and I_t is the time derivative of the brightness intensity[3].Thus optical flow can give significant information about the spatial arrangement of the objects viewed and the rate of change of this arrangement.

Multiple objects can be tracked easily from the video dataset as shown in Fig. 4.It shows original video first, then motion vectors calculated can be seen in second window. Third window showing objects segmented due to thresholding and the last window shows multiple objects being tracked.

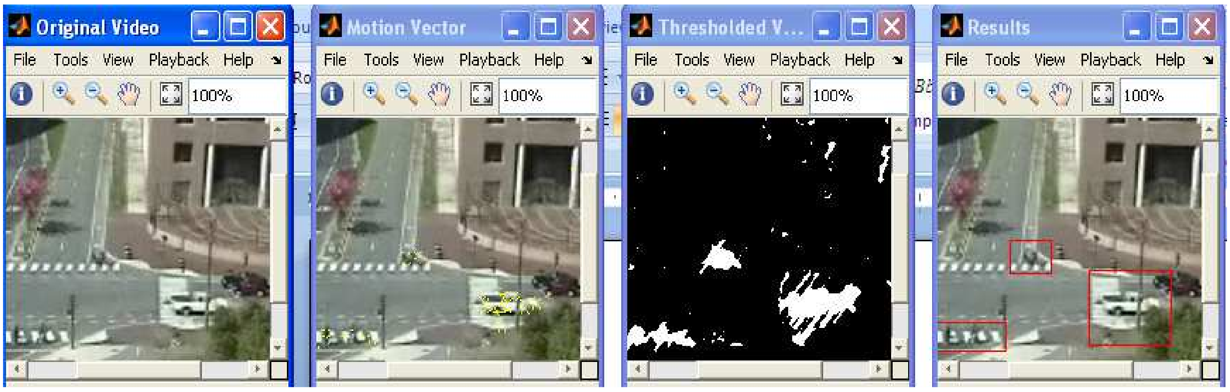

Fig. 4 Multiple object tracking using optical flow from random frame

IMPROVED OPTICAL FLOW ALGORITHM FOR OCCLUSION HANDLING

While optical flow algorithm tracks multiple objects but it is failed to track if there is change in intensity of moving objects. It was not able to track occluded objects also. We have improved existing optical flow algorithm for handling occlusion.

Improved optical flow algorithm for Occlusion handling
Input V ← Video file
1. Extract horizontal and vertical components of optical flow with varied frame difference delay.
For linear function:- $A_i x + A_j y + A_k = 0$ (15)
Where A_i, A_j and A_k are the spatiotemporal image brightness derivatives
x is the horizontal optical flow and y is the vertical optical flow.
(x,y,∇t) where ,∇t is frame difference delay.
2. µ ← compute mean for each frame // Find the mean for each frame
3. V_{med}← median () // Apply median filter for removal of noise
4. Apply morphological close and erosion operation on each frame
5. **For each** $frame_i$ ∈ Video
6. Estimate optical flow. The optical flow vectors are stored as complex numbers. Compute their magnitude squared which will later be used for thresholding.

$$E = \iint (A_i x + A_j y + A_k)^2 didj + \propto \iint \{(\frac{\partial x}{\partial i})^2 + (\frac{\partial x}{\partial j})^2 + (\frac{\partial y}{\partial i})^2 + (\frac{\partial y}{\partial j})^2\}\, didj \quad (16)$$

In this equation, $\frac{\partial x}{\partial i}$ and $\frac{\partial x}{\partial j}$ are the spatial derivatives of the optical velocity component *x*, and α scales the global smoothness term. E is estimation of optical flow.

7. Compute the velocity threshold from the matrix of complex velocities.

$$x_{i,j}^{m+1} = x_{i,j}^{-m} - \frac{A_i(A_i x_{i,j}{}^{-m} + A_j y_{i,j}{}^{-m} + A_k)}{\propto^2 + A_i{}^2 + A_j{}^2} \quad (17)$$

$$y_{i,j}^{m+1} = y_{i,j}^{-m} - \frac{A_j(A_i x_{i,j}{}^{-m} + A_j y_{i,j}{}^{-m} + A_k)}{\propto^2 + A_i{}^2 + A_j{}^2} \quad (18)$$

In this equation, $[x_{i,j}^m, y_{i,j}^m]$ is the velocity estimate for the pixel at (*x*, *y*), and $[x_{i,j}^{-m}, y_{i,j}^{-m}]$ is the neighbourhood average of $[x_{i,j}^m, y_{i,j}^m]$. For *m=0*, the initial velocity is 0.

8. If $frame_i \leq \sigma$// σ ← Threshold the image.

End for

9. $Thin\_frame_i$=morph_thin ($frame_i$) //Apply thinning to the Objects to fill the holes in the blobs.
10. Compute area → function_area ($frame_i$) // Estimate the area and bounding box of the blobs.

$$Area = (x_{max} - x_{min}) * (y_{max} - y_{min}) \quad (19)$$

11. Draw bounding boxes around the tracked objects.
12. Calculate and draw the motion vectors.

$$d_x = x' - x = f_x(a, x, y), d_y = y' - y = f_y(b, x, y) \quad (20)$$

Where x, y-location in previous image, x', y'-location in current image a, b- motion vector coefficient and d_x, d_y- displacement
13. Display results with tracked videos.

Hence tracking of objects from a given video dataset has become efficient as we are able to track more number of objects as well as occlusion can also be handled during the tracking as shown in Fig.5

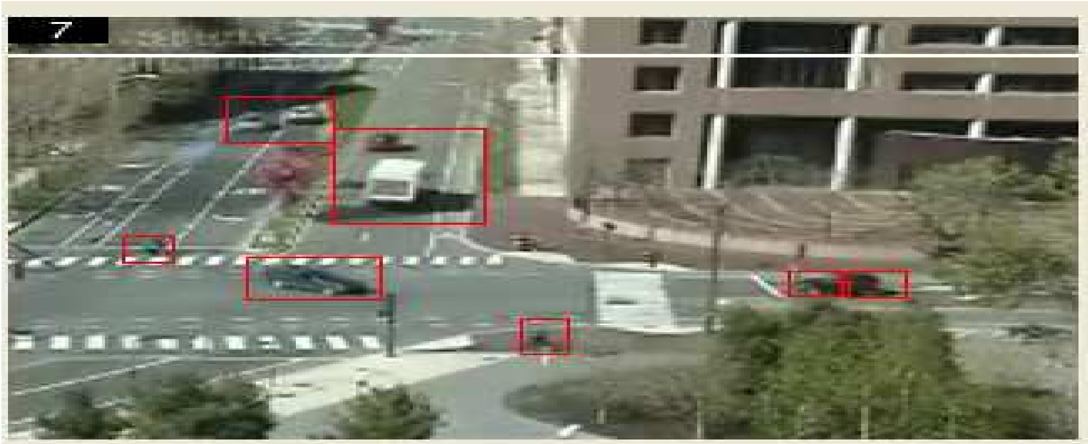

Fig. 5 Multiple object tracking using optical flow under occlusion

PERFORMANCE ANALYSIS

We observe the performance of the algorithms under various conditions on MIT Traffic video dataset as shown in Table 1. We observed that kalman filter algorithm is good for detecting moving object; moderate for lighting changes, roped frames, similar background shapes, similar background colours but it is bad for dim light video dataset, low resolution videos, stationary object and change in velocity. Whereas optical flow algorithm is good for detecting moving object, similar background shapes, similar background colours and change in velocity; moderate for dim, bright video datasets and dropped frames but it performs bad for low resolution video and lighting changes. Improved optical flow algorithm is moderate with dim, bright, dropped frames and lighting changes. It is bad only for stationary objects but good for moving objects with similar background shapes and colours and also good if there is change in velocity. The tracker has been tested under some tracking videos. To evaluate the performance of our method we use the accuracy measure. Tracking methods can be evaluated on the basis of whether they are able to track objects in a video. Given a ground truth, the performance can be evaluated by computing accuracy measure. In the context of tracking, the accuracy can be defined as:

$$\text{Accuracy} = \frac{\text{Correctly Tracked}}{\text{Total objects}} \qquad (21)$$

Correctly tracked is the number of objects which are detected and tracked correctly by the tracking algorithm. And the total Objects term is the total number of objects in video. Another parameter time is also tested for tracking algorithms. Accuracy of algorithms is shown in Table 2 and also graphical form in Fig. 6.

Table -1 Performance Analysis of Kalman, Optical Flow and Improved Optical Flow Tracking

Algorithm	Kalman Filter	Optical Flow	Improved optical flow
dim	bad	Ok	Ok
bright	Ok	Ok	Ok
Lighting changes	Ok	bad	Ok
Moving object	good	good	good
Stationary object	bad	bad	bad
Dropped frames	Ok	Ok	Ok
Low resolution	bad	bad	Ok
similar back ground shapes	Ok	good	good
Similar back ground colours	Ok	good	good
Change in Velocity	bad	good	good

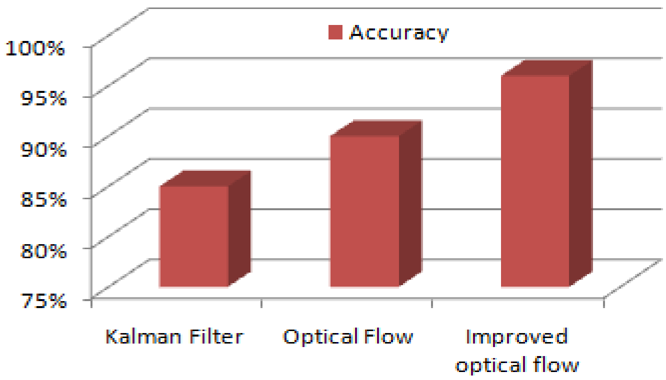

Fig. 6 Graphical representation of accuracy of algorithms

Table -2 Accuracy of Kalman, Optical Flow and Improved Optical Flow Tracking

Algorithm	Accuracy	Time required to track objects
Kalman Filter	85%	3-5 minutes
Optical Flow	90%	2-3 minutes
Improved optical flow	96%	1-2 minutes

CONCLUSION

In this paper we presented a tracking method for processing video data in order to perform tracking by a machine vision system. It summarizes as multiple objects can be tracked simultaneously. It can control some problem of multi object tracking, such as appearance and disappearance of objects, and missing of an object .Amongst the methods reviewed, improved optical flow algorithm is found to be more promising as it gives better accuracy in less computation time. The present work can be extended further by using more tracking algorithms and comparing their performance accordingly to achieve more accuracy. Also we will try to test our algorithm on real time videos.

REFERENCES

[1] CY Liu, P L Shui and S Li, Unscented Extended Kalman Filter for Target Tracking, *Journal of Systems Engineering and Electronics*, **2011**, p.188-192.

[2] B Horn and B Schunck, Determining Optical Flow Artificial Intelligence, *World Academy of Science, Engineering and Technology*, **1981**, 16,185–203.

[3] Aslani Sepehr and Nasab Homayoun Mahdavi, Optical Flow Based Moving Object Detection and Tracking for Traffic Surveillance, *International Journal of Electrical, Computer, Electronics and Communication Engineering*, **2013**, Vol 7 No 9.

[4] J Berclaz, F Fleuret, E Turetken and P Fua, Multiple Object Tracking using K-Shortest Paths Optimization, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2011** , p. 1806-1819.

[5] Y Zhai, MB Yeary, S Cheng and N Kehtarnavaz, An object-tracking algorithm based on multiple-model particle filtering with state partitioning, *IEEE Transactions on Instrumentation and Measurement*, **2009**, p. 1797-1809.

[6] Rakshit S and Anderson CH, Computation of Optical Flow using Basis Functions, *IEEE Transactions on Image Processing*, **1997,** vol 6, no 9, p.1246-1254.

[7] BD Lucas and T Kanade, An Iterative Image Registration Technique with an Application to Stereo Vision, *Proceeding of Imaging Understanding Workshop*, **1981**, p. 121-130.

[8] A Agarwal, S Biswas, S Singh, S Sural and AK Majundar, Object Tracking using Background Subtraction and Motion Estimation in MPEG Videos, Springer-Verlag Berlin Heidelberg, ACCV, **2006**, p.121-130.

[9] Qiaona Pei, Moving Objects Detection and Tracking Technology based Optical Flow, *North China University of Technology*, **2009,** p. 11-14.

[10] Zervos Michalis, Fua Pascal and Shitrit Horesh Ben, Real Time Multi-Object Tracking using Multiple Cameras, *Semester Project, School of Computer and Communication Sciences*, **2012.**