# Real-time Human Detection and Tracking Using a Very Low Resolution Camera

**M. Rezaur Rahman, M. Jahidul Islam, Bruce Poon, M. Ashraful Amin, Hong Yan**

***Abstract*— This work presents a robust and computationally efficient method for human detection and tracking. The unique feature of this method is that it has dedicated threads for human detection and camera control for human tracking. Moreover, it works with infra-red on and infra-red off. The method consists of five parts – training image acquisition, background subtraction, feature extraction, system training and system testing. Firstly, some sample video clips have been taken with an IP camera for initial system implementation. The clips are then being filtered for separating background and foreground. After that, some morphological operations are carried out to identify the most significant motion in the foreground. Those parts are cropped with some extra area and used to train a multiclass support vector machine (SVM) along with an image subset of the people detection dataset of The National Institute for Research in Computer Science and Control (French: *Institut National de Recherche en Informatique et en Automatique*, INRIA). A total of 597 images have been used as positive images and a total of 662 images have been used as negative images. Average detection accuracy of the system without infra-red is 89.37% and average detection accuracy of the system with infra-red is 72.66%. Therefore the average detection accuracy is 81.1%. We conclude (using dependent probabilistic analysis) that our system performs on an average of 89.37% accuracy based on our frame based analysis of video feeds.**

***Index Terms*-- Human tracking, Computer vision, Surveillance, Background subtraction, HOG**

## I. INTRODUCTION

Video surveillance has been a very important security measure throughout the world for quite some time. In some countries, it is imperative to have video surveillance in places like streets, shops, shopping malls, hospitals, parking lots etc. However, these systems always need human supervision for pan-tilt-zoom (PTZ) operations and they are ineffective for any sort of notification in case of any significant event without any human intervention.

In Bangladesh, video surveillance is becoming more and more popular with coming days. However, people are backing out of this idea as not only the camera setup for the system is costly but it also carries on a system-lifelong cost for maintenances and manpower. Our idea is to create a sustainable system that cuts the maintenance costs to the minimum and in most cases cuts the manpower cost totally. With the advancement of technologies in the field of computer vision, it is very possible to do so today.

The system elucidated in this paper has an intelligent which not only detects human subjects in the camera's field of view (FOV) but also does PTZ operations based on the movements of the subjects. To reduce processor and memory usage the system runs the human detection and PTZ operations in different threads. The system scans through the continuous video feed from the camera and starts tracking the human subject as soon as the subject enters the field of view of the camera. When the subject nears an edge (left or right) the camera will start the PTZ operations. The block diagram in Figure 1 shows the input and output of the system.

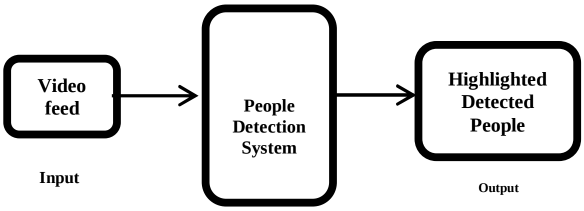

**Figure 1.** A simple block diagram of the system

## II. RELATED WORKS

A substantial amount of work has been done on human detection. Many different approaches have been taken by different researchers. Navneet Dalal and Bill Triggs [1] studied the question of feature sets for robust visual object recognition - adopting linear SVM based human detection as a test case. After reviewing existing edge and gradient based descriptors, they showed experimentally that grids of Histograms of Oriented Gradient (HOG) descriptors significantly outperformed existing feature sets, in their case the MIT pedestrian database, for human detection. Wang et al. [2] combined Histograms of Oriented Gradients (HOG) and Local Binary Pattern (LBP) as the feature set, and proposed a novel human detection approach capable of handling partial

This work is supported by G4S Security Services Bangladesh (P) Ltd (http://www.g4s.com.bd/).

M. Rezaur Rahman is with the Computer Vision & Cybernetics Research, SECS, Independent University, Bangladesh, Bashundhara, Dhaka 1229, Bangladesh (e-mail: refayetbd@gmail.com)

M Jahidul Islam is with the Computer Vision & Cybernetics Research, SECS, Independent University, Bangladesh, Bashundhara, Dhaka 1229, Bangladesh (e-mail: jahid1081@gmail.com).

Bruce Poon is with the School of Electrical & Information Engineering, University of Sydney, NSW 2006, Australia (e-mail: bruce.poon@sydney.edu.au).

M. Ashraful Amin, is with the Computer Vision & Cybernetics Research, SECS, Independent University, Bangladesh, Bashundhara, Dhaka 1229, Bangladesh. (e-mail: aminmdashraful@ieee.org).

Hong Yan is with the Department of Electronic Engineering, City University of Hong Kong, Hong Kong, China (e-mail:h.yan@cityu.edu.hk)

**ISBN: 978-0-9803267-5-8**

occlusion. Two kinds of detectors, i.e., global detector for whole scanning windows and part detectors for local regions, were learned from the training data using linear SVM. Viola and Jones [3] introduced a new image representation called the "Integral Image" which allowed the features used by the detector to be computed very quickly. On those images they used a learning algorithm, based on AdaBoost, which selected a small number of critical visual features from a larger set and yields extremely efficient classifiers [4]. Sabzmeydani and Mori [5] introduced an algorithm for learning shapelet features, a set of mid–level features. These features were focused on local regions of the image and were built from low–level gradient information that discriminated between pedestrian and non–pedestrian classes. Andriluka et al. [6] combined the advantages of both detection and tracking in a single framework. The approximate articulation of each person was detected in every frame based on local features that model the appearance of individual body parts. Prior knowledge on possible articulations and temporal coherency within a walking cycle were modelled using a hierarchical Gaussian process latent variable model (hGPLVM). Yao and Odobez [7] modelled their method based on a cascade of LogitBoost classifiers relying on features mapped from the Riemanian manifold of region covariance matrices computed from input image features. Finally, Zhu et al. [8] used AdaBoost for feature selection and a cascade of HOG to detect humans.

## III. The Proposed System

The complete methodology of our system is represented in figure 2 as a flowchart showing every step and its sequence. The individual steps are modularized and are often autonomous and sometimes dependent on each other.

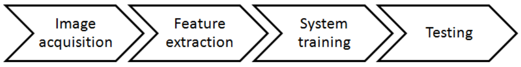

**Figure 2.** A flowchart of the methodology of our system

### A. Image Acquisition and Data Collection

Image acquisition is the first and one of the most essential tasks. Without a substantial number of images of numerous people from different angles the proposed method would not be much useful. The easiest option for acquiring images of people is using digital cameras. For our purpose, high quality image is not necessary at all. Instead, proper acquisition of the image is much more important. That is why we used an infrared IP camera with networking capabilities, namely Foscam FI8918W Wireless IP Camera [9].

### B. Sample Information

Proper image acquisition is very important. There are several things that should be kept in mind while taking photo of a human body using digital cameras.

- **Number of images:** The dataset can be partitioned into two groups. First one includes 1059 images from the INRIA database. The second part of the dataset contains 200 images acquired by us. Images in this partition were captured using infra-red mode of the wireless IP camera. In total there are 1400 images. Figure 3 shows a glimpse of our dataset. The following criteria were kept in consideration for optimal data selection.
- **Background:** The subject should be in front of solid background and the color of the background should not match the color of the clothing of the subject. For example, a white wall can be an excellent background.
- **Lighting:** As we are using an infra-red camera, use of any extra lighting, other than the existing and natural lighting, should be avoided to have any unwanted illumination causing a big white area in the image.
- **Camera Level:** The camera should be kept at head level of the subject so that even if the subject gets much closer to the camera we would be able to get the image of at least half of the body.
- **Resolution:** Resolution of the image is not a big factor. However, if the resolution is too low, for example, less than 600 x 400 pixels with the subject occupying less than 40% of the pixels, the chances are the method will not perform as expected. On the contrary, if the captured image is too big, there is absolutely no problem as long as the image is taken properly.

  Figure 3 shows an example of images captured with our infra-red IP camera using a white wall background and a door.

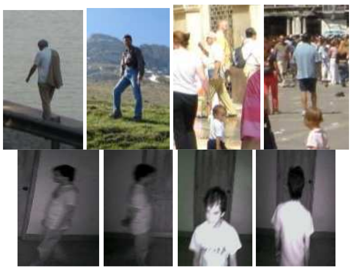

**Figure 3.** First row shows some of the images taken from the INRIA dataset and second row shows examples of some images captured using our camera

### C. Feature Extraction

There have been many different approaches over the years for extracting features for human detection. Dalal and Triggs first described Histogram of Oriented Gradient (HOG) descriptors in their June 2005 paper to the CVPR (conference on Computer Vision and Pattern Recognition) [1]. However, they applied their method on large images. We hypothesized that their method would yield more accurate results if we applied it on a smaller area. Therefore, we decided to do a background subtraction first between two consecutive frames to find out the difference between them and then applied the HOG descriptor on the subtracted part to determine whether there was a human in it.

#### *C.1. Background Subtraction:*

Background subtraction is a central component of many computer vision systems, used for detecting moving objects in videos. The main idea of this approach is that of detecting the moving objects from the difference between the current frame and a reference frame and threshold the results to generate the objects of interest. Existing methods for background modeling may be classified as either predictive or non-predictive.

Predictive methods model the scene as a time series and develop a dynamical model to recover the current input based on past observations. The second class of methods (called non-predictive density based methods) neglects the order of the input observations and builds a probabilistic representation of the observations at a particular pixel. For our purpose we used the mixture of Gaussians method for background subtraction.

Sometimes the changes in the background object are not permanent and appear at a rate faster than that of the background update. Typical examples of high frequency changes in scene are trees' leaves, snow, and rain or sea waves. In these cases, a single-valued background is not an adequate model. Stauffer and Grimson [10] raised the case for a multi-valued background model able to cope with multiple background objects. The authors describe the probability of observing a certain pixel value, $x$, at time $t$ by means of a mixture of Gaussians:

$$P(x_t) = \sum_{i=I}^{K} \omega_{i,t} \eta(x_t - \mu_{i,t}, \sum{}_{i,t}) \tag{1}$$

with each of the K Gaussian distributions deemed to describe only one of the observable backgrounds or foreground objects. In practical cases, K is set to be between 3 and 5. Gaussians are multi-variant to describe red, green and blue values.

The discrimination between foreground and background is achieved like this: Firstly, all the distributions are ranked based on the ratio between their peak amplitude, $\omega_i$, and standard deviation $\sigma_t$. The assumption is that the higher and more compact the distribution, the more is likely to belong to the background. After that, the first B distributions in ranking order satisfying:

$$\sum_{i=1}^{B} \omega_i > T \tag{2}$$

with T an assigned threshold, are accepted as background.

At each $t$ frame time, two problems must be simultaneously solved: a) assigning the new observed value, $x_t$, to be the best matching distribution and b) estimating the updated model parameters. These concurrent problems can be solved by an expectation-maximization (EM) algorithm. However, as this would prove extremely costly, the matching is approximated in these terms: amongst all distributions satisfying

$$\frac{x_t - \mu_t}{\sigma_{i,t}} > 2.5 \tag{3}$$

The first in ranking order is accepted as a match for $x_t$. Furthermore, probability density function (pdf) parameters ($\mu_{i,t}$,$\sigma_{i,t}$,$\omega_t$) are updated only for this matching distributions. If no match is found, the last ranked distribution is replaced by a new one centered in $x_t$ with low weight and high variance.

In the case where the background has very high frequency variations, this model fails to achieve sensitive detection. Modeling the background variations with a small number of Gaussians distributions will not be accurate and the very wide background distribution will result in poor detection.

### *C.2. Histogram of Oriented Gradients (HOG)*

The essential thought behind the Histogram of Oriented Gradient descriptors is that local object appearance and shape within an image can be described by the distribution of intensity gradients or edge directions. The implementation of these descriptors can be achieved by dividing the image into small connected regions, called cells, and for each cell compiling a histogram of gradient directions or edge orientations for the pixels within the cell. The combination of these histograms then represents the descriptor. For improved accuracy, the local histograms can be contrast-normalized by calculating a measure of the intensity across a larger region of the image, called a block, and then using this value to normalize all cells within the block. This normalization results in better invariance to changes in illumination or shadowing.

### *C.3. Algorithm implementation of HOG*

- *Gradient Computation:* The first step of calculation in many feature detectors in image pre-processing is to ensure normalized color and gamma values. However, Dalal and Triggs pointed out that this step could be omitted in HOG descriptor computation as the ensuing descriptor normalization essentially achieved the same result. Image pre-processing thus provided little impact on performance. Instead, the first step of calculation is the computation of the gradient values. The most common method is to simply apply the 1-D centered, point discrete derivative mask in one or both of the horizontal and vertical directions. This method specifically requires filtering the color or intensity data of the image with the following filter kernels:

$$[-1,0,1] \text{ and } [-1,0,1]^T \tag{4}$$

- *Orientation Binning:* The second step of calculation involves creating the cell histograms. Each pixel within the cell casts a weighted vote for an orientation-based histogram channel based on the values found in the gradient computation. The cells themselves can either be rectangular or radial in shape, and the histogram channels are evenly spread over 0 to 180 degrees or 0 to 360 degrees, depending on whether the gradient is “unsigned” or “signed”. Dalal and Triggs found that unsigned gradients used in conjunction with 9 histogram channels performed best in their human detection experiments. As for the vote weight, pixel contribution can either be the gradient magnitude itself, or some function of the magnitude. In actual tests, the gradient magnitude itself generally produces the best results.
- *Descriptor Blocks:* In order to account for changes in illumination and contrast, the gradient strengths must be locally normalized, which requires grouping the cells together into larger, spatially connected blocks. The HOG descriptor is then the vector of the components of the normalized cell histograms from all of the block regions. These blocks typically overlap, meaning that each cell contributes more than once to the final descriptor. Two main block geometries exist: rectangular R-HOG blocks and circular C-HOG blocks. R-HOG blocks are generally square grids, represented by three parameters: the number of cells per block, the number of pixels per cell, and the number of channels per cell histogram. In the Dalal and

Triggs human detection experiment, the optimal parameters were found to be 3x3 cell blocks of 6x6 pixel cells with 9 histogram channels. Moreover, they found that some minor improvement in performance could be gained by applying a Gaussian spatial window within each block before tabulating histogram votes in order to weight pixels around the edge of the blocks less. The R-HOG blocks appear quite similar to the scale-invariant feature transform descriptors. However, despite their similar formation, R-HOG blocks are computed in dense grids at some single scale without orientation alignment, whereas SIFT (Scale-invariant feature transform) descriptors are computed at sparse, scale-invariant key image points and are rotated to align orientation. In addition, the R-HOG blocks are used in conjunction to encode spatial form information, while SIFT descriptors are used singly. C-HOG blocks can be found in two variants: those with a single, central cell and those with an angularly divided central cell. In addition, these C-HOG blocks can be described with four parameters: the number of angular and radial bins, the radius of the center bin, and the expansion factor for the radius of additional radial bins. Dalal and Triggs found that the two main variants provided equal performance, and that two radial bins with four angular bins, a center radius of 4 pixels, and an expansion factor of 2 provided the best performance in their experimentation

- *Block Normalization:* Dalal and Triggs explored four different methods for block normalization. Let $v$ be the non-normalized vector containing all histograms in a given block, $\|v\|k$ be its *k*-norm for $k=1,2$ and $e$ be some small constant (the exact value, hopefully, is unimportant). The normalization factor can be one of the following:

  L2-norm: $$f=\frac{v}{\sqrt{\left(\|v\|_2^2+e^2\right)}} \quad (5)$$

  L2-hys: L2-norm followed by clipping (limiting the maximum values of v to 0.2) and renormalizing, as in

  L1-norm: $$f=\frac{v}{\left(\|v\|_1+e\right)} \quad (6)$$

  L1-sqrt: $$f=\sqrt{\frac{v}{\left(\|v\|_1+e\right)}} \quad (7)$$

  In addition, the scheme L2-Hys can be computed by first taking the L2-norm, clipping the result, and then renormalizing. In their experiments, Dalal and Triggs found the L2-Hys, L2-norm, and L1-sqrt schemes provided similar performance, while the L1-norm provided slightly less reliable performance. However, all four methods showed very significant improvement over the non-normalized data.
- *SVM Classifier:* The final step in object recognition using Histogram of Oriented Gradient descriptors is to feed the descriptors into some recognition system based on supervised learning. The Support Vector Machine classifier is a binary classifier which looks for an optimal hyperplane as a decision function. Once trained on images containing some particular object, the SVM classifier can make decisions regarding the presence of an object, such as a human being, in additional test images. In the Dalal and Triggs human recognition tests, they used the freely available SVMLight software package in conjunction with their HOG descriptors to find human figures in test images.

### *C.4. Feature Specifications for human tracking*

Firstly, we have used Shi-Tomasi corner detection [11] to determine good features to track.

Without loss of generality, we will assume a grayscale 2-dimensional image is used. Let this image be given by $I$. Consider taking an image patch over the area $(u,v)$ and shifting it by $(x,y)$. The weighted *sum of squared differences* (SSD) between these two patches, denoted $S$, is given by:

$$S(x,y)=\sum_u\sum_v w(u,v)\left(I(u+x,v+y)-I(u,v)\right)^2 \quad (8)$$

$I(u+x,v+y)$can be approximated by a Taylor expansion. Let $I_x$ and $I_y$ be the partial derivatives of $I$, such that

$$I(u+x,v+y)\approx I(u,v)+I_x(u,v)x+I_y(u,v)y \quad (9)$$

This produces the approximation

$$S(x,y)\approx\sum_u\sum_v w(u,v)\left(I_x(u,v)x+I_y(u,v)y\right)^2 \quad (10)$$

which can be written in matrix form:

$$S(x,y)\approx\begin{pmatrix}x & y\end{pmatrix}A\begin{pmatrix}x\\ y\end{pmatrix} \quad (11)$$

where *A* is the structure tensor,

$$A=\sum_u\sum_v w(u,v)\begin{bmatrix}I_x^2 & I_xI_y\\ I_xI_y & I_y^2\end{bmatrix}=\begin{bmatrix}\langle I_x^2\rangle & \langle I_xI_y\rangle\\ \langle I_xI_y\rangle & \langle I_y^2\rangle\end{bmatrix} \quad (12)$$

This matrix is a Harris matrix, and angle brackets denote averaging (i.e. summation over $(u,v)$). If a circular window (or circularly weighted window, such as a Gaussian) is used, then the response will be isotropic.

A corner (or in general an interest point) is characterized by a large variation of $S$ in all directions of the vector $\begin{pmatrix}x & y\end{pmatrix}$. By analyzing the eigenvalues of $A$, this characterization can be expressed in the following way: $A$ should have two "large" eigenvalues for an interest point. Based on the magnitudes of the eigenvalues, the following inferences can be made based on this argument:

1. If $\lambda_1\approx 0$ and $\lambda_2\approx 0$ then this pixel has no features of interest.
2. If $\lambda_1\approx 0$ and $\lambda_2$ has some large positive value, then an edge is found.
3. If $\lambda_1$ and $\lambda_2$ have large positive values, then a corner is found.

Harris and Stephens [12] noted that exact computation of the eigenvalues was computationally expensive since it required the computation of a square root. They suggested the following function $M_c$, where $k$ is a tunable sensitivity parameter:

$$M_c = \lambda_1\lambda_2 - k(\lambda_1 + \lambda_2)^2 = \det(A) - k \times trace^2(A) \quad (13)$$

The Shi-Tomasi [11] corner detector directly computes $\min(\lambda_1, \lambda_2)$ because under certain assumptions, the corners are more stable for tracking.

The value of $k$ has to be determined empirically. In the literature, values in the range 0.04 - 0.15 have been reported as feasible.

The covariance matrix for the corner position is $A^{-1}$, i.e.

$$\frac{1}{\langle I_x^2\rangle\langle I_y^2\rangle - \langle I_x I_y\rangle^2}\begin{bmatrix} \langle I_y^2\rangle & -\langle I_x I_y\rangle \\ -\langle I_x I_y\rangle & \langle I_x^2\rangle \end{bmatrix} \quad (14)$$

We have used the Lucas-Kanade optical flow method [13] which assumes that the displacement of the image contents between two nearby instants (frames) is small and approximately constant within a neighborhood of the point *p* under consideration.

Thus the optical flow equation can be assumed to hold for all pixels within a window centered at *p*. Namely, the local image flow (velocity) vector $(V_x, V_y)$ must satisfy

$$\begin{aligned} I_x(q_1)V_x + I_y(q_1)V_y &= -I_t(q_1) \\ I_x(q_2)V_x + I_y(q_2)V_y &= -I_t(q_2) \\ &\vdots \\ I_x(q_n)V_x + I_y(q_n)V_y &= -I_t(q_n) \end{aligned} \quad (15)$$

where, $(q_1, q_2, \ldots, q_n)$ are the pixels inside the window, and $I_x(q_i), I_y(q_i), I_t(q_i)$ are the partial derivatives of the image $I$ with respect to position *x*, *y* and time *t*, evaluated at the point $q_i$ and at the current time.

These equations can be written in matrix form

$$Av = b \quad (16)$$

where

$$A = \begin{bmatrix} I_x(q_1) & I_y(q_1) \\ I_x(q_2) & I_y(q_2) \\ \vdots & \vdots \\ I_x(q_n) & I_y(q_n) \end{bmatrix} \quad (17)$$

$$v = \begin{bmatrix} V_x \\ V_y \end{bmatrix} \quad (18)$$

and

$$b = \begin{bmatrix} -I_t(q_1) \\ -I_t(q_2) \\ \vdots \\ -I_t(q_n) \end{bmatrix} \quad (19)$$

This system has more equations than unknowns and thus it is usually over-determined. The Lucas-Kanade method obtains a compromise solution by the least squares principle. Namely, it solves the 2×2 system

$$A^T Av = A^T b \quad (20)$$

or

$$v = \left(A^T A\right)^{-1} A^T b \quad (21)$$

where $A^T$ is the transpose of matrix $A$. That is, it computes

$$\begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} \sum_i I_x(q_i)^2 & \sum_i I_x(q_i)I_y(q_i) \\ \sum_i I_x(q_i)I_y(q_i) & \sum_i I_y(q_i)^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum_i I_x(q_i)I_t(q_i) \\ -\sum_i I_y(q_i)I_t(q_i) \end{bmatrix} \quad (22)$$

with the sums running from *i*=1 to *n*.

The matrix $A^T A$ is often called the structure tensor of the image at the point *p*.

### D. System Training and Testing

Our HOG Descriptor had the following parameters–

- Window size – 64X128
- Block size – 16X16
- Block stride – 8X8
- Cell size – 8X8
- Bins – 9
- Sigma – minus one (-1)
- Threshold – 0.2

This configuration gave us 3780 features per image.

We used a two-class linear SVM to train our human detection system. The parameters that were used for our SVM are–

- Kernel – LINEAR
- SVM Type –Multi-class SVM
- Class – 2
- Termination criteria type – Iterative
- Number of Iterations – 2000
- Epsilon (required accuracy) – 0.000001

597 images of people (positive images) and 662 images of irrelevant objects (negative images) were used to train the system.

## IV. Result Analysis And Discussion

There are 2 segments of validation and testing that have been done rigorously and they are:

1. The Human "Detection" Accuracy Evaluation (separately analyzed with and without Infrared (IR) capabilities.

2. The "Detected Human Direction" Accuracy Evaluation.

### A. The Human "Detection" Accuracy Evaluation

Figure 4 and Figure 5 represent our detection accuracy findings with and without infrared (I.R.) respectively.

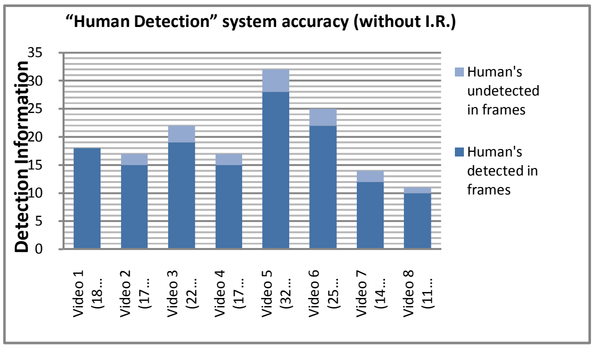

**Figure 4.** "Human Detection" system accuracy (without IR)

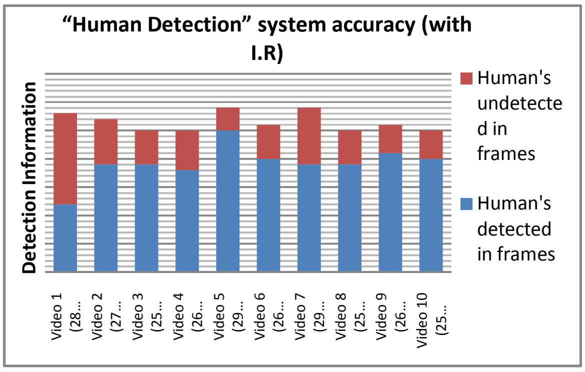

**Figure 5.** "Human Detection" system accuracy (with IR)

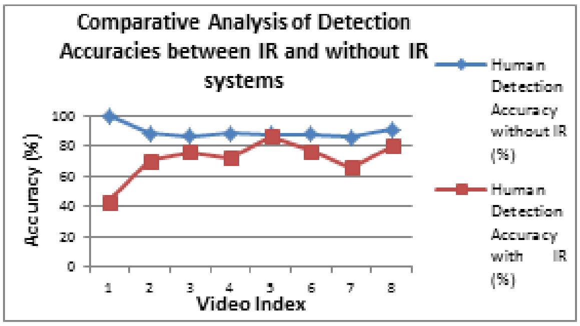

**Figure 6.** Comparative analysis of Detection accuracies between IR based and IR-less system.

Figure 6 illustrates a comparative analysis on the system when we enabled infrared capabilities and when we did not.

### B. The Average "Detection" Accuracy Evaluation

Average detection accuracy of system without IR = 89.37% (From Figure 4)

Average detection accuracy of system with IR = 72.66% (From Figure 5)

Therefore the average detection accuracy = (89.37+72.66)/2 = 81.1%

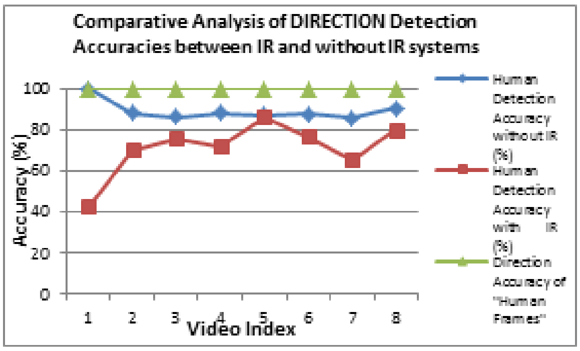

**Figure 7.** The Accuracy Analysis of Direction Detection of "Detected Humans" in reference to Detection Performance.

### C. The Accuracy Analysis of Direction Detection of "Detected Humans"

The second phase of our system was to track the movement of the detected humans in the videos. Apparently, according to our analysis and testing, we got proper direction in all the detected frames that were detected as "human frames". Thus it had remained constant at 100% in all the detected frames.

The plot in figure 7 shows the performances of both "detection" and "direction" accuracies. It should be noted that the direction accuracy are strictly based on "detected human frames."

### D. Overall Performance Accuracy of the System (Dependent Accuracy Analysis)

Since there are two dependent segments of the system and they give separate performance accuracies, it is imperative to generate an overall performance accuracy of the system. The 2 segments of the systems are:

A. Human Detection in the videos
B. Detected Human Movement Direction Tracking in the videos

If we notice closely, we see that part B is dependent on the performance of part A. Therefore, we can compute the average accuracy of the overall system consisting both A and B using conditional probability theory. It is axiomatic that "if A happens, then and only then B takes place" or in other words "the performance of B is meaningful based on the performance of A."

Therefore, P(A) = 89.37 % (considering "without IR" as we have got better accuracy there).

Considering B is dependent on A, P(B|A) = 100 % as B happens every time when A happens.

Considering B as independent, we again find, P(B) = 100 %

Thus overall performance of system, using Bayes' Theorem, P

$$(A\,|B) = \frac{P(B\,|\,A)\times P(A)}{P(B)} = \frac{100\times 89.37}{100}\% = 89.37\%$$

Therefore, we conclude that our system performs on an average of 89.37% accuracy based on our frame based analysis of video feeds.

## V. CONCLUSION

In this work, we present a robust and computationally efficient method for human detection from live video feed and tracking. Our system has three major functional units. The task of the first unit is to subtract background to identify any significant motion. The second unit deals with the identification of a human being within that window of significant motion. The third unit identifies the direction of motion of the human being. The unique feature of our system is that it has dedicated threads for human detection and camera control for human tracking. Moreover, it works with infra-red on and infra-red off. Overall, 1259 simple images, with 1059 of those being taken from the INRIA database and the rest acquired by us, have been used to train our system. Average detection accuracy of the system without infra-red is 89.37% and average detection accuracy of the system with infra-red is 72.66%. Therefore, the average detection accuracy is 81.1%. We conclude (using dependent probabilistic analysis) that our system performs on an average of 89.37% accuracy based on our frame based analysis of video feeds.

## REFERENCES

[1] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In Proc. CVPR, pages 886–893, 2005.

[2] X. Wang, T. X. Han, and S. Yan. An HOG-LBP human detector with partial occlusion handling. In Proc. of ICCV, 2009.

[3] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In Proc. CVPR, pages 511–518, 2001.

[4] Y. Freund and R. E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” J. Comput. Syst. Sci., vol. 55, pp. 119–139, 1997.

[5] P. Sabzmeydani and G. Mori. Detecting pedestrians by learning shapelet features. In CVPR, 2007.

[6] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection & people-detection-by-tracking. In CVPR, 2008.

[7] J. Yao and J. M. Odobez, “Fast human detection from videos using covariance features,” IDIAP Research Institute, Tech. Rep. 07-68, 2007.

[8] Q. Zhu, M. C. Yeh, K. T. Cheng, and S. Avidan. Fast human detection using a cascade of histograms of oriented gradients. In CVPR, pages 1491–1498, 2006.

[9] Foscam FI8918W (Black) Wireless IP Camera. Foscam Corporation. Retrieved March 20, 2012.
http://foscam.us/products/foscam-fi8918w-wireless-ip-camera-11.html

[10] C. Stauffer and W.E.L. Grimson. Adaptive background mixture models for real-time tracking. In CVPR, 1998.

[11] C. Tomasi and J. Shi. Good features to track. In CVPR94, pages 593–600, 1994.

[12] C. G. Harris and M. Stephens. A combined corner and edge detector. In 4th Alvey Vision Conference, pages 147–151, 1988.

[13] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In Proceedings of the 7th International Joint Conference on Artificial Intelligence, 1981.