Turk J Elec Engin, VOL.13, NO.2 2005, © TÜBİTAK

Silhouette Based Human Motion Detection and Analysis for Real-Time Automated Video Surveillance

Murat EKİNCİ, Eyüp GEDİKLİ
Dept. of Computer Engineering, Karadeniz Technical University,
Trabzon, 61080, TURKEY
e-mail: ekinci@ktu.edu.tr

Abstract

In this paper, a real-time background modeling and maintenance based human motion detection and analysis in an indoor and an outdoor environments for visual surveillance system is described. The system operates on monocular gray scale video imagery from a static CCD camera. In order to detect foreground objects, first, background scene model is statistically learned using the redundancy of the pixel intensities in a training stage, even the background is not completely stationary. This redundancy information of the each pixel is separately stored in an history map shows how the pixel intensity values changes till now. Then the highest ratio of the redundancy on the pixel intensity values in the history map in the training sequence is determined to have initial background model of the scene. A background maintenance model is also proposed for preventing some kind of falsies, such as, illumination changes (the sun being blocked by clouds causing changes in brightness), or physical changes (person detection while he is getting out or passing in front of the parked car). At the background modeling and maintenance, the reliability and computational costs of the algorithm presented are comparatively discussed with several algorithms. Based on the background modeling, candidate foreground regions are detected using thresholding, noise cleaning and their boundaries extracted using morphological filters. Then for people detection, object detection and classification approach for distinguishing a person, a group of person from detected foreground objects (e.g., cars) using silhouette shape and periodic motion cues is performed. Finally, the trajectory of the people in motion and several motion parameters produced from the cyclic motion of silhouette of the object under tracking are implemented for analyzing people activities such as walking and running, in the video sequences. Experimental results on the different test image sequences demonstrate that the proposed algorithm has an encouraging real-time background modeling based human motion detection and analysis performance with relatively robust and low computational cost.

Key Words: *Background model initialization and maintenance, motion detection, human motion analysis, real time vision, video surveillance.*

1. Introduction

Automated video surveillance is currently one of the most important research topics in the computer vision community. The development in this research area is being propelled by the increased availability of inexpensive computing power and image sensors, as well as the inefficiency of manual surveillance and

monitoring systems [1, 2, 3, 26, 27]. Nevertheless, mounting video cameras in the surveillance area is cheap, but finding available human resources to observe the surveillance area is expensive. Although, video cameras are already mounted in different environments such as banks, stores, monitoring human and vehicle traffics, *etc*, video data is currently stored to be use only "after the fact" as a forensic tool. Therefore it has emerged developments on partially or fully automate the task of surveillance for applications such as event detection, object classification, human action recognition. To be successful, such applications require real time motion detection and human motion analysis algorithms, which provide low-level functionality upon which higher level recognition capabilities can be built.

Existing automated surveillance systems can be classified into categories according to, the environment (*i.e.* indoor, outdoor), the number of sensors (*i.e* single camera vs. multiple cameras, color, gray scale). Large research studies devoted to automated video surveillance searches have been conducted in recent years [1, 2, 3, 4]. In addition, companies such as IBM and Microsoft are also investing on research on human motion analysis [5, 6]. One of the important research areas in the automated surveillance systems is automatically human behavior understanding [7, 33]. Automatically understanding human behavior from motion imagery involves extraction of visual information from a video sequence, representation of that information in a suitable form, and interpretation of visual information for the purpose of recognition and learning about human behavior. Biometrics is also a technology that makes use of the physiological or behavioral characteristics to authenticate the identifies of people [40, 8].

The initial stage of the extraction of visual information is the detection of moving objects from a video sequence. At the video surveillance researches, background subtraction techniques are mostly used for detection motion in many real-time video surveillance applications [1, 2, 3, 9]. Recently, there has been a large amount of work addressing the background subtraction, adaptation [1, 2, 15, 9, 11, 12], and background model initialization [13, 14]. Often the assumption is made that an initial background model can be obtained by using a short training sequence in which no foreground objects are present. However, in some monitoring areas, such as public area, crowded corridors, traffics, it is difficult or impossible to control the area being monitored. In such cases there may be needed to train the model using a sequence which contains foreground objects. An ideal background subtraction could produced good results while foreground regions in motions during training sequence. There is also needed to maintenance background model to adapt all possible changes in the monitoring area.

A large number of people detection and tracking algorithms rely on the process of background subtracting, a technique which detects changes from a model of the background scene. They mainly focused on model representation and techniques for adaptation on background modeling. The ability for representing multiple models to have the background values some techniques to model motion which is part of the background [14, 15]. Unimodal representations, which store a single mean intensity value per pixel, are more prone to have false detections in the case of moving any background regions. The methods in some studies estimate background intensity values using temporal smoothing [9, 12, 15], and choose a single value from the set of past observations [11]. In [2] minimum and maximum intensity values , and maximum temporal derivative for each pixel are stored for initialization background model. The study in [16] proposed an edge-based background representation called the background primal sketch. PFinder [12] uses a unimodal background model to locate interesting objects. In [17], it is presented an adaptive background mixture model for real-time tracking. In their work, they modeled each pixel as a mixture of Gaussian and used an on line approximation to update it. McKenna [30] used an adaptive background model combining color and gradient information, then background subtraction performed to cope with shadows and unreliable color

cues. In [17], an adaptive multi-modal background subtraction method that can deal with slow changes in illumination, repeated motion from background clutter and long term scene changes is employed.

After background subtraction the detected objects are tracked using multiple hypothesis tracker. Common patterns of activities are statistically learned over time and unusual activities are detected. W4 [2] uses dynamic appearance models to track people. A recursive convex hull algorithm is used to find body part locations for single person. Symmetry and periodicity analysis of each silhouette is used to determine if a person is carrying an object. Ricquerberg and Bouthemy [18] proposed tracking people by exploiting spatio-temporal slices. Their detection scheme involves the combined use of intensity, temporal differences between three successive images and of comparison of the current image to a background reference image which is reconstructed and updated on line.

The following process after successfully tracked the moving humans from one frame to another in an image sequence in the video surveillance applications is human behaviors understanding from image sequences. Human behavior understanding is to analyze and recognize the motion region segments by reason of human actions in frames and to produce high-level description of human actions. There has been considerable interest in the area of human motion analysis in recent years [1, 2, 7, 25]. Further works are also focused to human identification based on gait analysis [33, 34, 38, 31].

This paper is a significant extension of an earlier and much shorter version presented in [19]. The purpose and main contributions of this paper are as follows:

- We attempt to develop a simple but effective approach for human motion detection and analysis as a real-time video surveillance system.
- A new fast and robust algorithm for the purpose of background model initialization and maintenance is presented. In the model initialization, the algorithm takes as input a video sequence in which moving objects are included, and outputs a statistical background model describing the static parts of the scene.
- To be able have more reliable and fast algorithm for background estimation based human motion detection and analysis, a real-time background maintenance algorithm is developed for illumination changes, such as the sun being blocked by clouds causing changes in brightness, or physical changes such as deposited objects.
- Object detection and classification approach for distinguishing a person, a group of person from detected foreground objects (*e.g.*, cars) using silhouette shape and periodic motion cues is presented in the following section.
- Initial studies on human motion analysis for distinguishing walking and running actions are explained.
- One of the main objective of this paper is also to present a set of techniques integrated into a low-cost PC based real time visual surveillance system for simultaneously detection and tracking people, and monitoring their activities in monochromatic video.

The remainder of this paper is organized as follows: Section 2 describes algorithms to detect people in surveillance scenes. First, a statistical background model initialization is estimated using historical variations on each pixel locations in a training video sequence. This is extensively clarified in section 2.1. Secondly, a maintenance algorithm for adaptation of the background modeling to the variations in the scene by the time is explained in section 2.2. Then, the following section 2.3 gives details of how the foreground objects

detected are classified as a person, a group of person, or others. Section 3 focuses on tracking an isolated person using motion correspondence between regions that are moving in a 2D space. An approach for human motion analysis is also explained in section 4. Then, experimental results and discussion on future work are presented and deduced in section 5. Section 6 concludes the paper.

2. Human Motion Detection

Moving human detection is the first step processes for nearly every system of vision-based human analysis. The aim on moving human detection is to segment the regions corresponding to people from the rest of an image sequence. It is known to be a significant and difficult research problem [14]. There are three conventional approaches to moving object detection: temporal differencing (two-frame or three-frame) [20, 21, 23], optical flow [13], and background estimation [1, 2, 14, 9]. Temporal differencing is very adaptive to dynamic environments, but generally does a poor job of extracting all relevant feature pixels. Optical flow can be used to detect independently moving targets in the presence of camera motion, however most optical flow computation methods are very complex and are inapplicable to real-time algorithms without specialized hardware [13]. Background subtraction is a particularly popular method for motion segmentation especially under those situations with a relatively static background. It attempts to detect moving regions in an image by differencing between current image and a reference background image in a pixel-by-pixel fashion. However, it is extremely sensitive to changes of dynamic scenes due to lighting and extraneous events. To overcome that problems a new statistical background model initialize and a maintenance of the background model approach is presented in the following section.

2.1. Background Model Initialization

The initial background model is obtained even if there are moving foreground objects in the field of view, such as walking people, moving cars, *etc.* The basic idea in the background model initialization presented in this study is depending on that stationary pixel intensity value is brightness value which has the highest redundancy ratio on intensity values taken from a training sequence. Let F be an array containing T consecutive images, $F^i(x)$ is the intensity of a pixel location x in the i *th* image of F. The initial background model for a pixel location x, $h(x)$, is obtained as follows:

$$h(x) = Highest\_Redundant\{N(F^t(x))\} \tag{1}$$

Here, $N(F^t(x))$ indicates the number of the redundancy of the intensity value at pixel location x in all images in F.

To obtain the background model initialization, first, the number of the redundancy ratios of the intensities at pixel location x in all images of F is calculated to several seconds of video (typically 10-30 seconds) to distinguish moving pixels from stationary pixels. Training sequence is called to this video sequence. Each intensity values taken by the pixel location x in the training sequence is compared with the intensity values taken hitherto. Then the intensity variations of that pixel are also stored into a cell string assigned for the pixel location x in an history map. At the history map, is shown in Figure 1, the first cell in the string (is shown by the string labeled with **D**) stores the number of different intensity values taken for the pixel location x during the training sequence. The following two cells, shown in Figure 1, the first one labeled with **1a** stores an intensity value taken for the pixel location x, the other cell labeled with **1b** stores the number of the redundancy of the that intensity value taken for the pixel location x during the

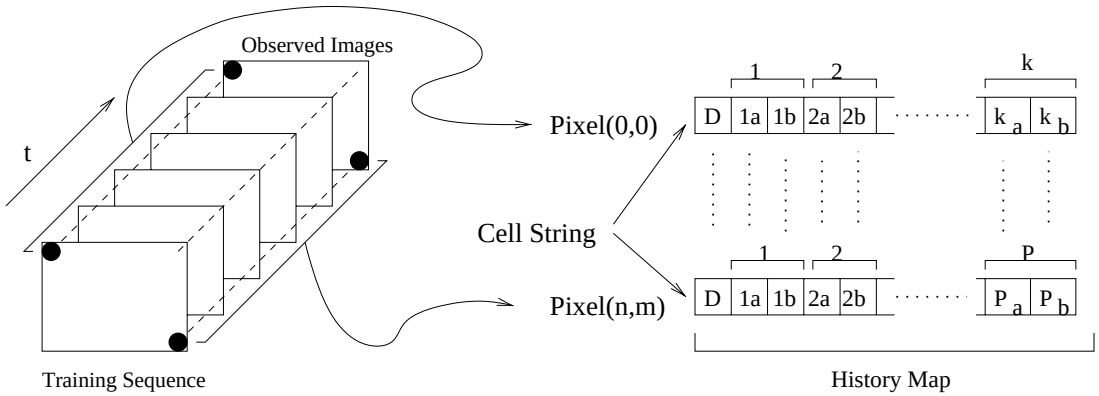


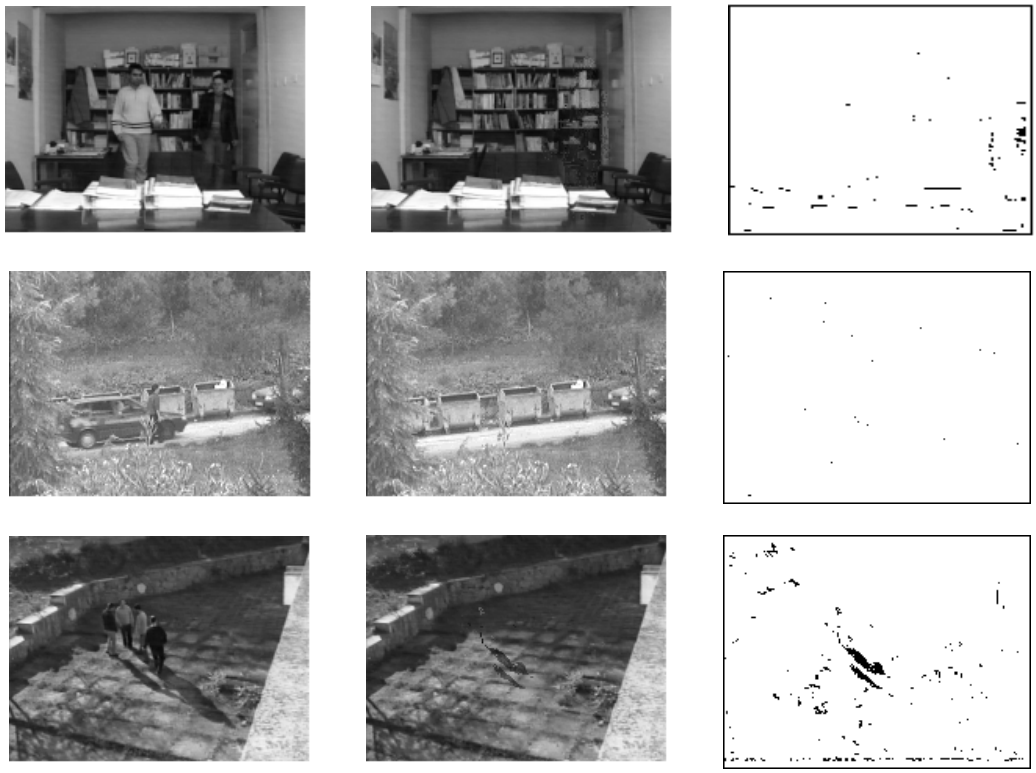
Figure 1. To obtain initial background model. **(Left)** Training sequence, (200-600 frames), **(Right)** Estimation of the history map during training sequence for each pixel.

training sequence. The following every other both cells in the cell string are also used to store same both data structure for other possible intensity values taken for the pixel location x and their redundancy numbers unless the pixel has an intensity value taken before. The size of the cell string for the pixel location x is dynamic and is also depending on the number of possible different intensity values taken at the pixel location in x during the training sequence. This is repeated for each pixel locations in the background model frame.

The algorithm for obtaining the history map is as follows: At the first frame, F^1, the all first cells, **D**, of the the strings in the history map are settled to 1 because the first image in the training sequence is in processing. The intensity value of a pixel location x in the first image is also stored into the first sub cell, **1a**, of the cell string for the pixel location x in the background model. Then the other sub cell, **1b**, of the cell string is settled to 1 due to the first image in the processing. These all setting processes are done for every pixel locations in the first frame. At the second frame, F^2, if the current intensity value of the pixel location x is same with the intensity value stored before for the pixel location x (the string cell labeled with **1a**) in the history map, the value in the sub-cell, **1b**, is one-incremented. Therefore, the value in the cell segment, **D**, is not changed because the current intensity value at the pixel location x is not different with the value stored before in the sub-cell **1a**. In other possibility, if the new current intensity value of the pixel location x is different with intensity value stored in the sub-cell **1a**, the size of the cell string for the pixel location x is extended to store the current intensity value and its redundancy number. That is, the different intensity value and its initial redundancy value settled to 1 are stored in the sub-cells labeled with **2a**, and labeled with **2b** of the extended cell respectively, because of the new intensity taken. The value in the cell **D** for the pixel location x is one incremented owing to differencing on the intensity taken in the pixel location x, and so on. This is done for each pixel location in the frame. Then the all processes are repeated for every frames in the training sequence.

As a result, the all data (intensity value and its redundancy value) stored for each pixel location in the history map are linked with each other to represent the history of the training sequence. At the end of the learning stage, the highest redundancy ratio (HRR) for the pixel location x is estimated using the data stored for the pixel location x in the history. Then the intensity value linked with the HRR for the pixel location x is assigned as the stationary pixel intensity value for the location x in the initial background model. This is done for estimation of the all stationary pixel intensity values on the pixel locations in the background model.

The output of our algorithm for several example provided in different training sequences is shown in Figure 2. At the top in Figure 2.a, two people are traveling in a room, at the middle, a crowded traffic includes people and cars is being on the street in the university campus, at the bottom, multiple people are moving in outdoor some of them together others separate. The results of the HRR algorithm to produce the initial background model on the related scenario are shown in Figure 2.b. In Figure 2.c, the map of error pixels between ideal background and the estimated background model obtained by the HRR algorithm are shown.



(a) (b) (c)

Figure 2. Results of running algorithm on three different sequences: (a) a snapshot of the training sequence, (b)the computed background model, (c) map of error pixels (black).

The idea of the HRR algorithm developed is inspired from median filter. The advantages of the HRR algorithm to the other algorithms depend on the median filter are;

- Use of the median relies on an assumption that the background at every pixel will be visible more than fifty percent of the time during the training sequence. But, in the algorithm presented, the assumption for the background visibility at the every pixel is becoming to the highest redundancy ratio of the intensity values during the training sequence. This is more flexible and applicable for real events than median filters and also includes all advantages of the algorithms depending on median filter process.
- The requirement memory size for applying the HRR algorithm is more less than the others based on median filter.
- The realization processing time of the HRR algorithm is also rather less because there is no need the sorting of the intensity values taken during training and maintenance sequences after end of that sequences.

As explained the advantages two of which very important for real time maintenance of the background are successfully provided advantages of the HRR algorithm. To make more clarify the advantages of the algorithm approved, the following test processes are explained.

(a) (b) (c)

Figure 3. Example frames in the scenario for testing the algorithm. (a) A person enters to the scene, waits, (b) and go away, (c) a second person enters, waits, and go away.

(a) (b) (c)

Figure 4. Background model initialization results produced by different algorithms on the training sequence in the scenario for testing the algorithm. The scenario in training sequence is same in Figures 3, 5. (a) Median-based algorithm (b) W4 algorithm (c) HRR-based algorithm presented.

To test the reliability of the background model initialization based on the HRR method a scenario was produced. The scenario in the video sequence, example frames are shown in Figure 3, is; a person is moving to a position in the scene, and waiting a period of time, then going away from the position. After followed by a period of time no one is at the position, then another person who is becoming reason to producing different intensity has done similar activity in the scene. Figure 5 shows a plot of intensity over time for one pixel in the video sequence (200 frames) obtained from the scenario above. In addition lighting effects (there was no significant weather effects because of the video sequences obtained indoor) tend to cause smooth changes with no large steps. The pixel location under considered in the scene has mainly four different types of intensities belong to background ($\sim$%38), a person ($\sim$%28), other person ($\sim$%26), and a period of instability ($\sim$%8), respectively. Three algorithms ([2], median, HRR) are separately tested on the video sequence to determine background intensity for initial background model. The intensity is assigned for the pixel location as 68 by median-based, 56 by the W4[2], and 105 by the HRR-based algorithm. The real background intensity value for the pixel location was a value changing between 100 and 115 in the test sequence. The results on the background model initialization performed by three different algorithms (median-based, w4, HRR-based) are shown in Figure 4 for whole image. These results have been obtained on the test sequence includes 200 frames explained in Figure 5, and some example frames in training sequence are also shown in Figure 3.

The more experimental results on the different video sequences are shown in Figure 6. Three algorithms are especially applied to five seconds of video to obtain background model. The period of the

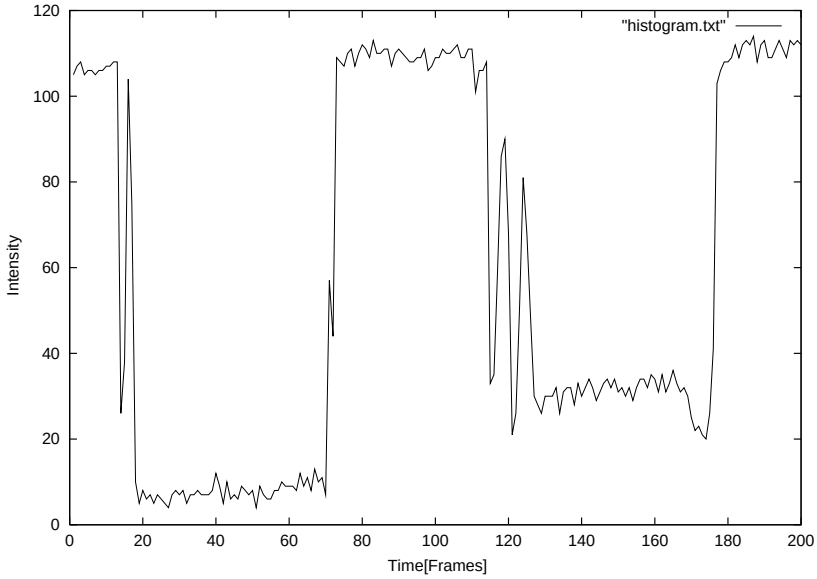
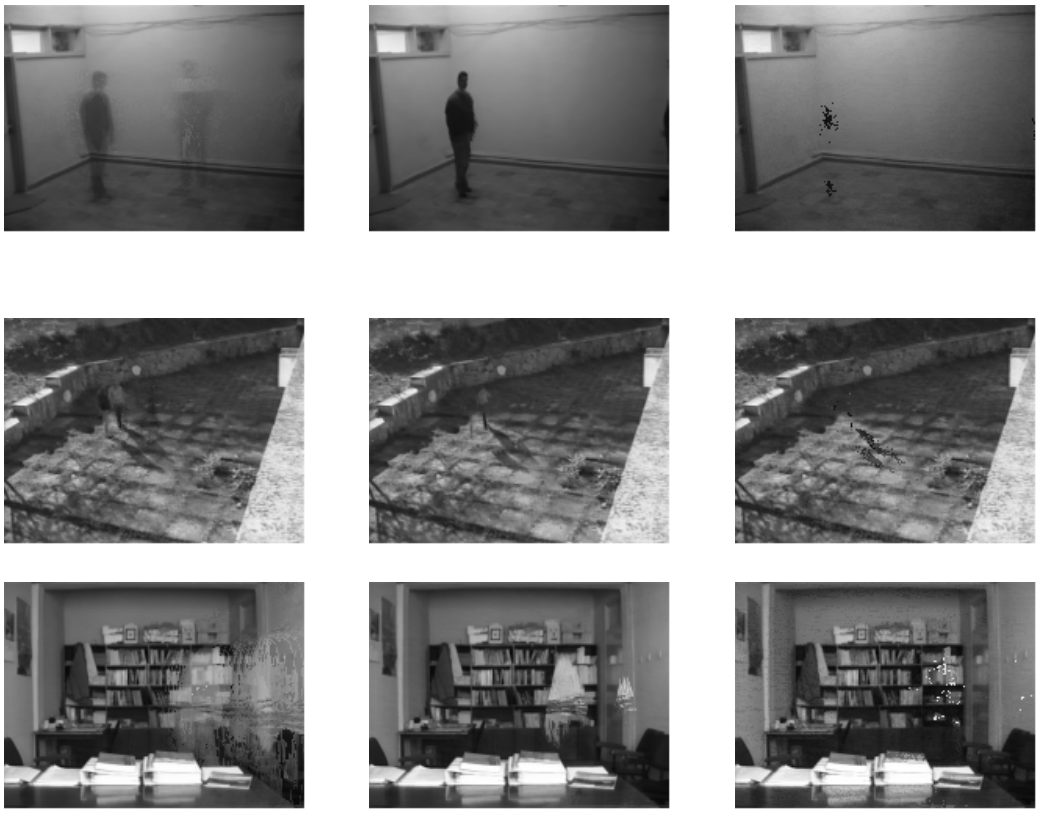


Figure 5. Example analysis of intensity changes over time at a single pixel as a person enters the scene, waits and go away, a second person enters, waits, and go away. Each of these steps in visible in the pixel's intensity profile.

learning has been short to be able more visibility on differencing between the algorithms. If the learning period was more long, the background model produced by each algorithms would big probability be more robust. Consequently, the HRR algorithm has produced encouraged results for more robust and more reliable approach on both short and long time of period in the test video sequences.

The requirement memory size for the background modeling is to be more less is another advantage of the HRR algorithm. One of the criteria on the size of the history map (see fig. 1) used by the HRR algorithm is the quantized ranges used to quantize the possible pixel brightness in the video sequence. In other words, the algorithm operates on monocular gray-scale video imagery. The pixel has 255 different intensity. If there is no any intensity quantized range for a pixel location in the image, a maximum memory size to store the history for the pixel location is 513 byte (1 byte for **D** segment, 512 bytes for each intensity and its redundancy number, see Figure 1). If there is an intensity quantized range, for instance 20, the maximum memory size is 25 byte (1 byte for **D** segment, 24 bytes for others). The results for the different intensity quantized ranges used are shown in Figure 7 for the processing time and the memory size needed. The results have been produced for 197x149 pixels in monocular images (200 frames) an indoor environment. At the indoor applications, the choosing of the optimum intensity quantized range, *i.e.* 5, is important for both the processing time and the memory size take into account the lighting effects tend to obtain on the reliable results.

Figure 8 shows the comparison results for two algorithms (HRR based presented, and median based). The experimental results shown in Figure 8, have been produced on two different scenario in the indoor and the outdoor environments. When the graphics in Figure 8 is considered, it is seen that the memory size used by the HRR based algorithm is depending on possibility of the number of different intensity variations rather than the number of the frame in the training sequence. The memory sizes needed for the HRR and



(a) (b) (c)

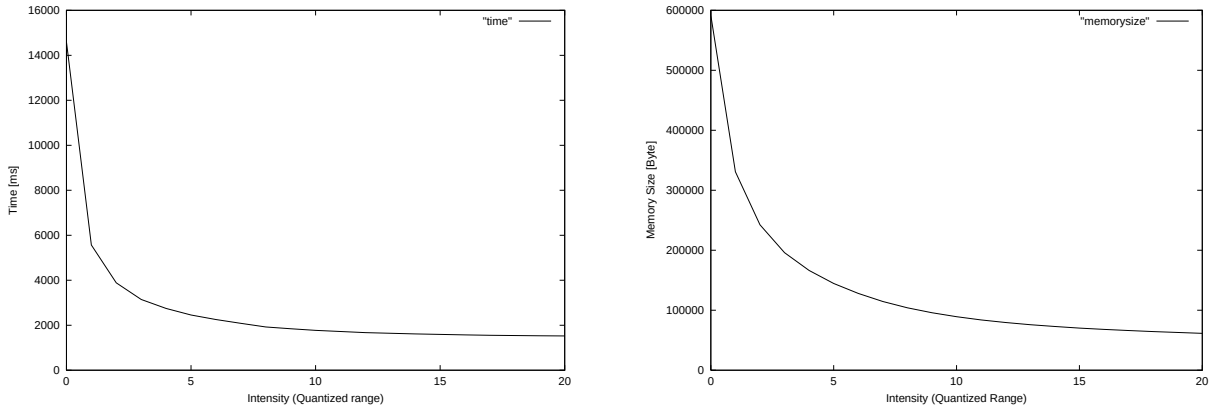
Figure 6. Results of three algorithms on three different sequences: (a) W4 algorithm, (b) Median based, (c) HRR algorithm presented.

median based algorithms may be calculated as follows, respectively:

$$\text{Memory\_Size} = \frac{2 * \text{ Frame\_Resolution } * 256}{\text{Brightness\_Quantized\_Range}} \tag{2}$$

$$\text{Memory\_Size} = \text{Frame\_Resolution } * \text{ Number\_of\_Frames} \tag{3}$$

The other advantage of the algorithm presented is the processing time for the background modeling. The classification of the intensity values in the history map for a pixel location in the background modeling is simultaneously performed by taking the training sequence from the CCD camera. Then the following process is easily to find the highest redundancy ratio from the value stored in the cell strings labeled with **b** in the history map (see Figure 1) for each pixel location. In the median based and W4 algorithms [2], the background modeling is simply performed by sorting intensity values taken for the pixel location then determined median value as a stationary pixel intensity. The all processes are done after the training sequence is moved to the memory, but in the algorithm presented the background modeling (producing of the history map) is simultaneously performed while the frame grabbing. This is highly reducing the time consuming on the median process for real-time applications of the background modeling and updating. Figure 9 shows comparative results for the background modeling processing time on the different algorithms (HRR based, median based, and W4 algorithms).



(a) (b)

Figure 7. The processing time and the requirement memory size for the different intensity quantized ranges for the HRR algorithm based initial background modeling. (a) Processing time, (b) requirement memory size.

Table 1. Computation times.

Frame	Resolution	Processing Time *[msec]*		
		HRR	Median	W4
50	197x149	571	2283	2595
800	197x149	31716	758420	760744

The processing time on the other both algorithm is increased all the more while training sequence includes more frames. But, at the HRR algorithm the computation time is very low and almost very close each other for the training sequence includes more frames. The W4 system not only stored minimum, maximum intensity values, and maximum temporal derivative for each pixel but also used the median approach to the background modeling [2]. Seeing that reason, the computation time for modeling is a little more than the median. The most important parameters on the time consuming for the background modeling is the number of the frames in the training sequence. But in our approach, most important parameter is not that, it is the possibility on the intensity variations for the pixel locations, because different intensity values are considered by the background modeling approach presented. In Figure 9, the time consuming results on the processing for the background modeling are shown on the training sequences includes different number of frames. Table 1 gives the computation times on the training sequences includes different number of frames. The initial background modeling on the training sequence includes 50 frames has been executed at 571 *msec* by our approach, 2283 *msec* by median approach, 2595 *msec* by the W4 algorithm. For the training sequence includes 800 frames, the modeling time was 31716 *msec*, 758420 *msec*, and 760744 *msec* by the algorithms, respectively. These algorithms are separately implemented in C++ and runs under the Windows 2K operating system at 850 MHz Celeron PC, 96/133 MByte/MHz RAM for 197x149 resolution gray-scale images. As a result, at the test platforms, our approach has presented more robust and more applicable results for real-time background modeling, especially maintenance for video surveillance applications.

2.2. Maintenance Background Model

The difficult part of background subtraction is not the differencing itself, but the maintenance of a background model -some representation of the background and its associated statistics [14]. Background maintenance is

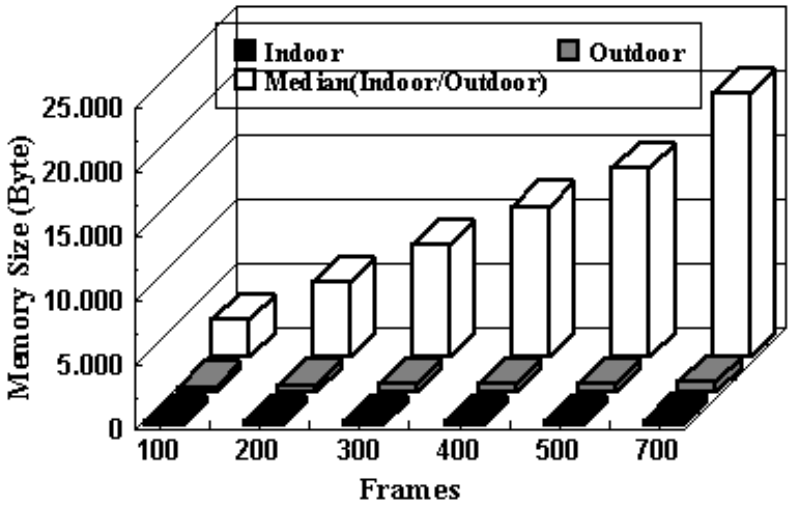


Figure 8. Comparison of memory sizes needed by two algorithms approach (median, HRR approved). The scenario at the indoor is two people travels in a room, at the outdoor, people, car, and some of groups includes people and cars are moving in the scene.

called to this modeling process. Also, real-time background maintenance is important for real surveillance applications. Because, the monitoring area may not be stay the same for long periods of time. There could be illumination changes, such as the sun being blocked by clouds causing changes in brightness, or physical changes, such as a car that comes into the scene and parks should not be considered as a part of the scene background, however the stationary pixels should play the role of background for detecting motion of a person getting out or passing in front of the car. What an ideal background maintenance system should have and the problems of background scene maintenance for surveillance system are good explained in [14].

In these cases, the notion in this study is to use an adaptive background model to accommodate changes to the background while maintaining the ability to detect independently moving objects (person(s)). The algorithm presented is based on two processes to update the background model.

- A *pixel-based maintenance*,
- A *object-based maintenance*.

The **pixel-based maintenance** updates the background model *sequentially* ($p_s(x)$) and *periodically* ($p_p(x)$) to adapt to illumination and physical changes in the background scene. The all processes at the pixel-based maintenance are performed at the low-level processing. In the surveillance scene as long as no tracking objects detected, the pixel-based maintenance is activated without considering a mid-level processing. In other words, if the current intensity on a pixel location is not background model value, the current intensity value is only considered by the periodically adaptation, otherwise the intensity is considered by both sequentially and periodically adaptations of the pixel-based maintenance.

The sequential adaptation of the pixel-based maintenance is dealt with by using a statistical model of the background to provide a mechanism to adapt to slow changes in the scene. This is performed using a temporal (low pass) filtering. The filter is implemented:

$$\overline{K}_{n+1} = \alpha * K_{n+1} + (1 - \alpha)\overline{K}_n \tag{4}$$

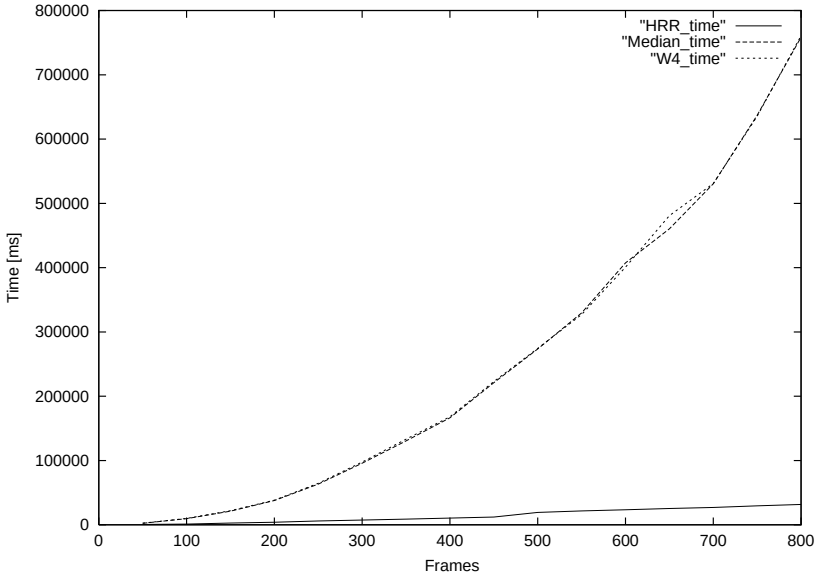


Figure 9. Comparison of time consuming by three algorithms approach (median, W4, HRR approved). The scenario at the indoor is two people travels in a room, at the outdoor, people, car, and some of groups includes people and cars are moving in the scene.

Where K_n is the value of each pixel in the n *th* frame, $\overline{K}_n$ is a running average, $\alpha = t * f$, t is a time constant which can be configured to refine the behavior of the system, and f is the frame rate.

The periodically adaptation of the pixel-based maintenance is also re-initialized the background model algorithm based on the HRR is applied on the pixel values in the periodical sequences (50-100 frames). The basic idea of the periodically adaptation is to adapt to high illumination and specially physical changes in the scene at the low level processing. Because a deposited/removed object, or a parked car would be added into the background scene if it does not move for along period of time. This idea has successfully produced robust results using only low-level processing for the maintenance of the background model.

The **object-based maintenance** updates the background model to adapt to physical changes originated by the object(s) in tracking in the background scene. In the object-based maintenance, if a pixel location is classified as a pixel location belongs to an object under tracking, the pixel-based maintenance is not enabled for the pixel position by the algorithm, otherwise enabled. In other words, this updating is a mid-level implementation whether the pixel-based maintenance is enabled for the pixel location or not.

In tracking, the background maintenance module constructs a decision map to determine whether a pixel-based maintenance or an object-based maintenance only, or both together are applied. The decision map represents the state of a pixel location which is classified as a background pixel, a foreground pixel, or an object pixel under tracking. Consequently, the background model pixel $p(x)$ is updated using the pixels which are classified as background pixels (p^b), the pixels which are classified as foreground pixels (p^f), and the pixels which are classified as object pixels under tracking (p^o). Let (p^c) be the background model currently being used; the maintenance background model pixels, $p(x)$, are determined as follows

$$p(x) = \begin{cases} p_p(x) & \text{if } x \text{ is foreground pixel, } p^f(x) \\ p_s(x) + p_p(x) & \text{if } x \text{ is background pixel, } p^b(x) \\ p^c(x) & \text{if x is object pixel under tracking, } p^o(x) \end{cases} \quad (5)$$

Where $p_p(x)$, $p_s(x)$ are periodically and sequentially pixel-based maintenance processes, respectively..

In Figure 10, initial background models, the models updated, and differencing on both are shown respectively. For the top Figure 10, no any object entered at the surveillance time. There was only the lighting effects in time for the scene. The sequentially pixel-based maintenance alone is good enough to update this type of scenario, so long as any object(s) does not enter into the scene. At the top-right picture in the Figure 10, the intensity variations caused by the lighting effects in time are shown as a map of error pixels (black). In the middle line of the Figure 10, a car entered into the scene then parked. To be able to detect any object(s) passing around the car, the intensities on the pixel locations occupied by the car should be added to the background model. The sequentially pixel-based maintenance is not activated for that pixel location because of an object detected. The HRR-based background updating (modeling), that is periodically pixel-based maintenance, is simultaneously processed on a period of the video sequence (100 frames). Therefore, the HRR algorithm automatically assigns the intensity values on the pixel locations occupied by the car as the background intensities updated.

When an object should be tracked by the system enters into the scene, the mid-level (object-based) background upgrading process is activated. An example frame in a scenario has this kind of properties is shown in the bottom pictures of the Figure 10. The mid-level process does not consider the pixel location belongs to the object for upgrading the background modeling, while the object is not moving or partially moving. That is, the object, for instance a person, first enters the scene, then stopped. After that (s)he is just moving his hand not other parts. The body parts not moving will be added to the background modeling unless the mid-level upgrading process is performed. The mid-level upgrading presented is developed to overcome this kind of problems.

In addition, sudden changes in background illumination, such as the lights indoor turning off or clouds blocking the sun, make the detection fail. Occasionally, it is possible to encounter similar catastrophes with this updating procedure and large parts of the image are classified as foreground and not object which is not detected. So, in that situation, the algorithm stops tracking and start to learn new background model, as described previously.

2.3. Foreground Region Detection and Classification

The basis idea on the foreground region detection based on adaptive background subtraction is to maintain a running statistical background value of the intensity at each pixel. The maintaining of that is explained at the previous sections. When the value of a pixel in a new image differs significantly from the background value, the pixel is flagged as potentially containing a foreground region. Then foreground objects are segmented from the background in each frame of the video sequence by a four stage process: thresholding, noise cleaning, morphological filters, and object detection. Each pixel is the first classified as either a background or a foreground pixel using the background model.

If a pixel has a value which is more than $2\sigma(\mathrm{x})$ from the background pixel intensity ($\overline{K}_n(\mathrm{x})$), then it is considered a foreground pixel. In here, $\sigma(\mathrm{x})$ is standard deviation of the largest inter frame absolute differences between the current images and the background images over the entire image, and $\overline{K}_n(\mathrm{x})$ is also explained in the previous section. The standard deviation value at pixel location x in all images is temporally upgraded using;

$$\overline{\sigma}_{n+1}(x) = \alpha(x)(K_{n+1}(x) - \overline{K}_{n+1}(x)) + (1 - \alpha(x))\overline{\sigma}_n(x) \tag{6}$$

where $\overline{\sigma}(x)$ running average of the standard deviation at pixel location x in all images.

Figure 10. The differencing between the background model updated and non-updated. (a) Non-Updated (Initial model), (b) Updated, (c) map of differentiation pixels (black).

Thresholding alone is not sufficient to obtain clear foreground regions; it results in a significant level of noise. The algorithm in this study uses region-based noise cleaning to eliminate noise regions, then foreground regions are filtered for size to remove spurious features. Next, the remaining regions are processed for distinguishing the object from other objects whether human or not. For that aim, the shape of a 2D binary silhouettes is initially represented by its projection histogram. The detection algorithm computes the 1D vertical and horizontal projection histograms of the silhouettes in each frame. Vertical and horizontal projection histograms are computed by projecting the binary foreground region on an axis perpendicular to the major axis and along the major axis, respectively. Using that projection histograms, the position of all foreground regions boundaries are detected as shown in Figure 11.

People have very distinctive shape, appearance, and motion patterns compared to other objects (car, animal, *etc.*). One can use a static shape analysis, such as aspect ratio, area, size perimeter, or dynamic motion analysis, such as speed, or periodicity of the motion to distinguish people from other objects. All cues in the static shape analysis can easily be obtained from the binary data produced by thresholding and the bounding box parameters, as shown in Figure 11. But the cues can be produced from the dynamic motion analysis presents more reliable information for classification, especially for the motion analysis. A skeleton based approach similar with [22] but more developed is presented in this study for the region classification and for the motion analysis.

While a foreground region is detected, there may be spurious pixels detected, holes in object features, 'interlacing' effects from video digitization process and so on. Therefore, the first pre-processing step to get

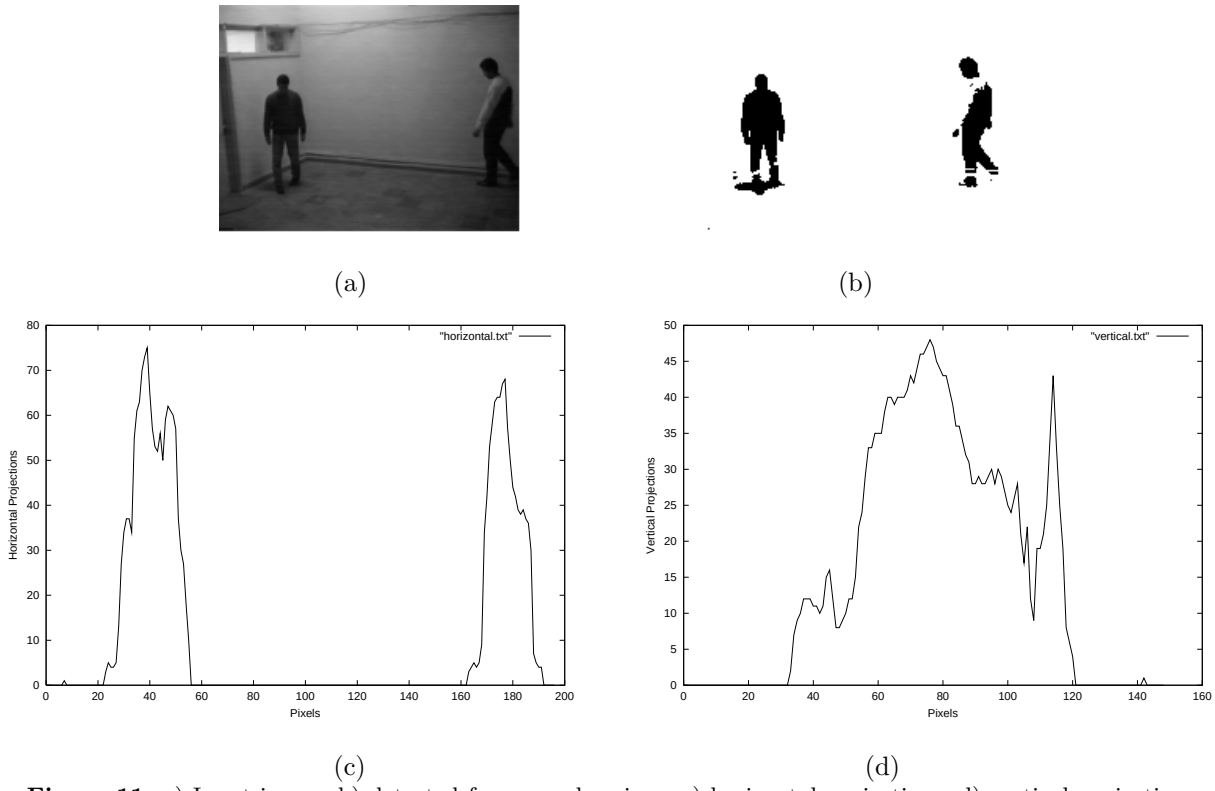


Figure 11. a) Input image, b) detected foreground regions, c) horizontal projections, d) vertical projections

skeletonization is to clean up anomalies in the detected regions. This is implemented by a morphological filter (dilation followed by an erosion). That is the silhouette region is dilated twice followed by a single erosion. This removes any small holes in the silhouette and smoothies out any interlacing anomalies. And then the outline of the silhouette region is extracted using a border following algorithm as shown in Figure 12.

The dynamic motion analysis process can be performed by determining the internal motion of a moving region over time. A good way to quantify the internal motion of the moving region is to use skeletonization. For skeletonization of the any shape there are many known standard techniques, such as thinning, and distance transformation. However, these standard techniques are computationally expensive and moreover, are highly susceptible to noise in the motion region boundary. The method presented similar to [22] and uses a simple real-time, robust way of detecting extremal points on the boundary of the motion region to produce its skeleton. The steps of this skeleton process are mainly shown in Figure 2.3 and illustrated as follows:

1. The center of the gravity of the foreground region boundary is determined,

2. The distances from the center of the region to each border point are calculated. Then the signal produced from the distances is smoothed to reduce the noise on the boundary. This can be done using a linear smoothing filter (*e.g.* mean, median), or a low pass filter in the time domain.

3. Local maximal of the filtered signal are obtained.

Finally, the local maximal of the signal represents the region boundary are taken as extremal points.

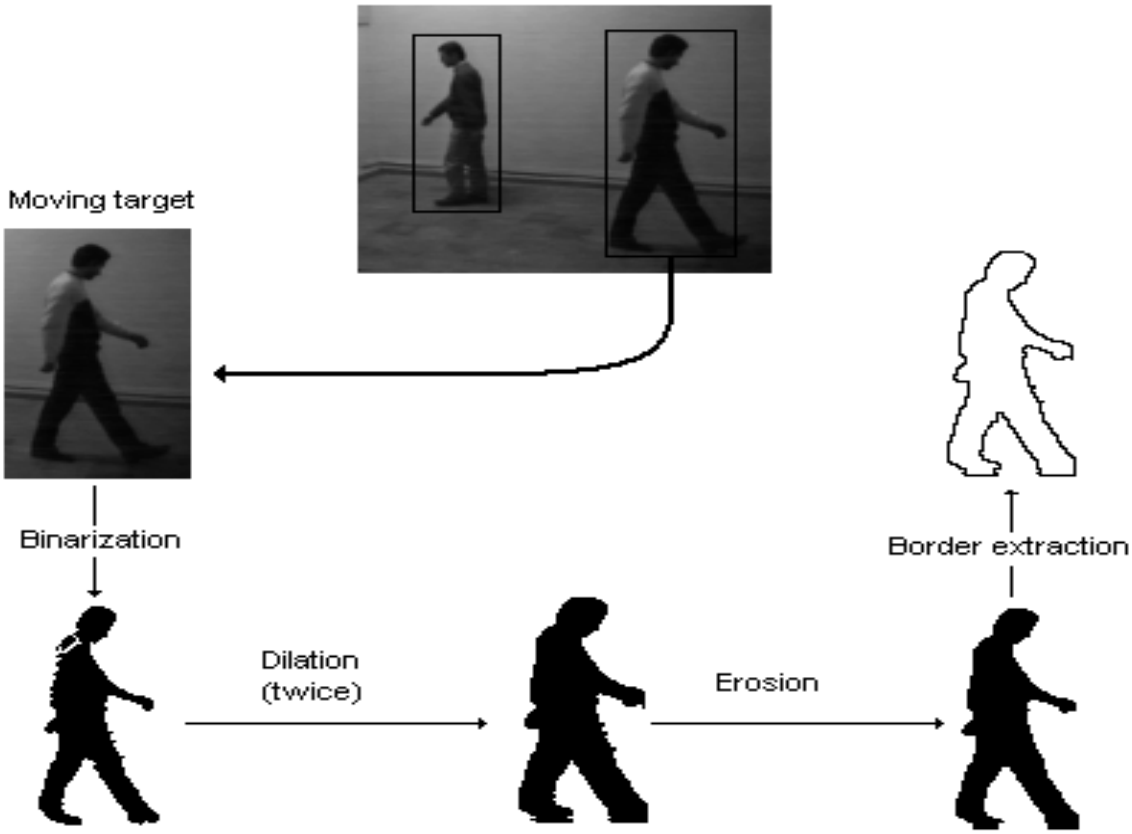


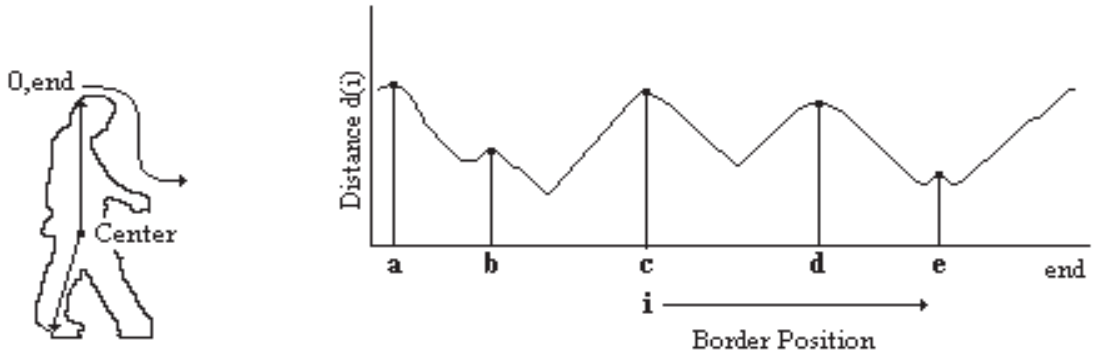
Figure 12. Foreground region pre-processing. A moving foreground region is morphologically dilated (twice) then eroded. Then its border is extracted.

Then the skeleton of the motion region is constructed by connecting them to the motion region centroid (x_c, y_c). This procedure for producing skeleton of the motion shape is illustrated in Figure 2.3.

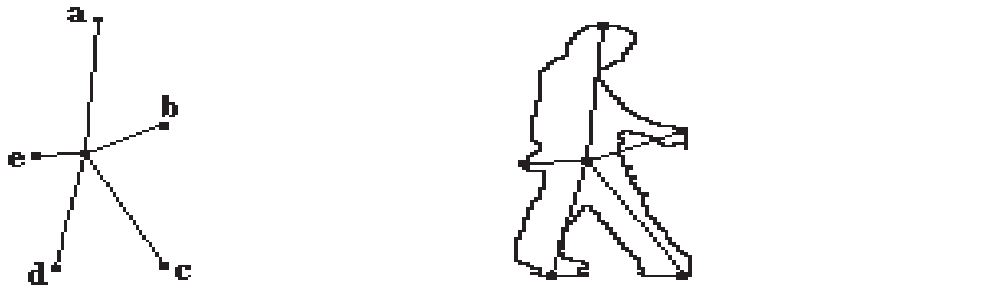
An example of the experimental results at different test platform for the producing the static and dynamic shape features is shown in Figure 14. The next goal in this section is to classify each moving object visible in the video sequence as a single person, a group of persons, or a vehicle. One of the advantages of video for classification is its temporal component. To exploit this, the static and dynamic features over time are computed in each bounding box as it is detected through the sequence of frames (3-10 frames).

The static shape features are directly measured from the silhouette and its bounding box. They are the bounding box aspect ratio, the axis of second moment of the silhouette, the bounding box dispersedness $(perimeter^2/area)$. The dynamic shape features are also produced from the skeleton of the silhouette of the region detected over time. Both the structure of the skeleton and repetitive changing on the skeleton gives important cues in analyzing different types of objects. Figures 15, 16, 17 show the structure of the skeleton for different types of objects (a person, a group of person, and a car). It is considered, the repetitive changing on the structures of the skeletons has another good enough feature for classification of the detected foreground regions in the image sequence.

The local maximal of the distances (see Figure 2.3) are determined as the structure of the skeleton. Then this feature can be used for human model criteria for deciding whether the object detected is human



(a) The center of the gravity. (b) The distances from the center to each border point.

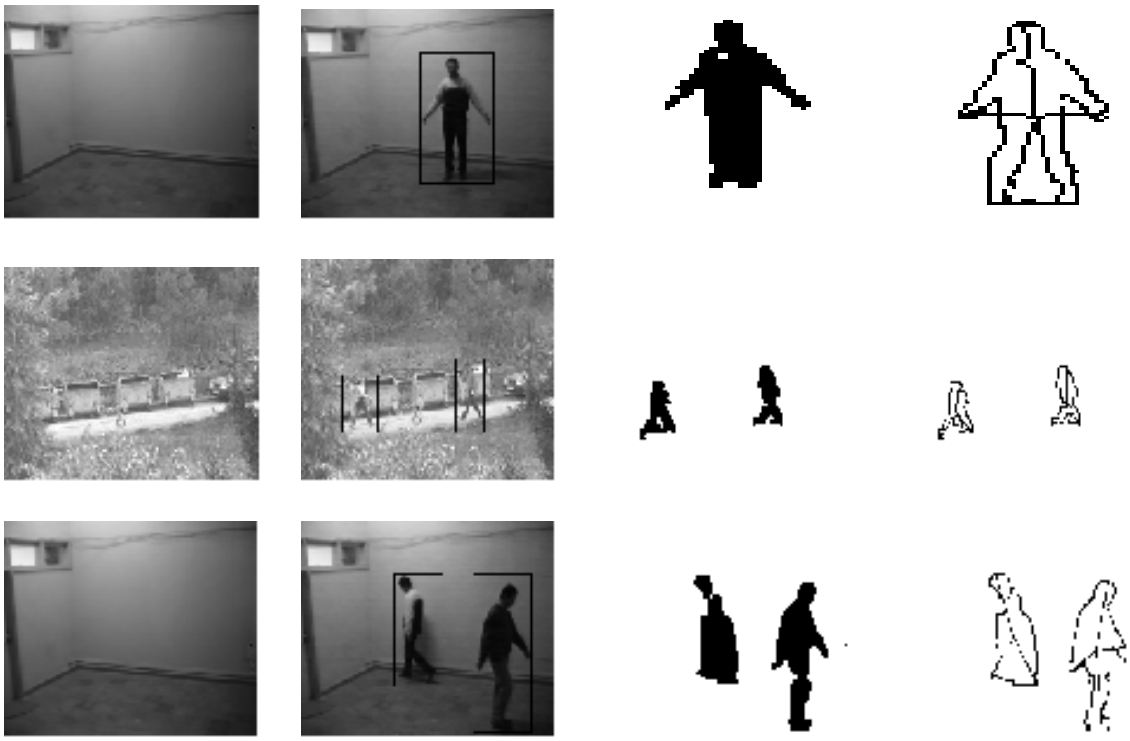


(c) local maximal points. (d) local maximal points projected onto the silhouette.

Figure 13. The region skeleton is created from silhouette boundary as a distance function from centroid, and extracting external points.

or not as shown in Figures 15, 16, 17. To more reliable classification another feature in the natural motion of people is also considered. It is that people exhibit periodic motion while they are moving. A periodic motion can be determined by self-similarity of the characteristics of silhouettes over time using the skeleton features of the silhouettes. As a result, static shape cues with a dynamic periodicity analysis and a periodicity of the motion analysis can be combined to distinguish human from the objects, such as a group of person, a car. The Figures 15, 16, 17 show three different objects in an image sequence and their skeletons produced. When the structure of skeletons in that Figures are considered, it is also clear that the structure and rigidity of the skeleton are important cue in analyzing different type of motion regions. These informations can also be used to understand human activities in the scene as explained in the following section.

As a summary, the process in this section is to classify each moving object visible in the video sequence as a single person, a group of persons, or a vehicle using the static shape analysis and the dynamic motion analysis. The classification results are produced after a short video sequence (3-10 frames) is processed. In this short video sequence, the detection algorithm provides the bounding box surrounding the silhouette of the foreground region, the skeleton of the silhouette, centroid and correspondence of each object over the frames.



(a) (b) (c) (d)

Figure 14. An example of foreground region detection while background has different intensity variation. a) upgraded background images, b) the current images, c) detected and binarization regions, d) skeletonization of the foreground regions.

3. Object Tracking

The goal of object tracking is to establish correspondence between objects across frames. It is assumed in this study that the regions can enter and exit the scene and they can also get occluded by other regions. Regions carry informations like shape and size of the silhouette, and colors data on a bounding box location estimated for each person.

Each region is defined by the 2D coordinates of the centroid, P, a ratio between the total number of foreground pixels *(T)* and the size of the bounding box *(B)*, $R = T/B$, and the color/gray level characteristic, D. The regions, for which correspondence has been established, have also an associated velocity, V. In frame t of a sequence, there are M regions with centroids P_i^t (where i number of regions) whose correspondences to previous frame are unknown. There are K regions with centroids P_L^{t-1} (where L is the label) in frame *t-1* whose correspondences have been established with the previous frames. The number of regions in frame t can be less than the number of regions in frame *t-1* due to entries and it might be less due to exits or occlusion.

The task is to establish correspondence between regions in frame t and frame *t-1*, and to determine entries and exits in these frames. The minimum cost criteria is used to establish correspondence. The cost function between two regions is defined as

$$C_{Li} = \frac{P_L^{t-1} + V_L^{t-1}}{P_i^t} + \frac{R_L^{t-1}}{R_i^t} + \frac{D_L^{t-1}}{D_i^t} \tag{7}$$

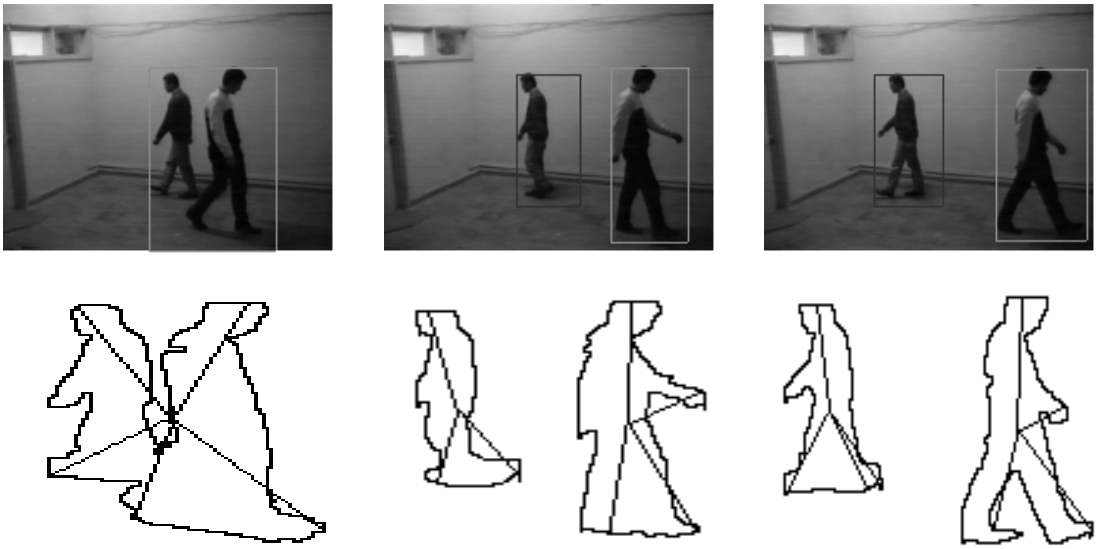


Figure 15. For the classification process on detected foreground regions, the dynamic structures of a single human.

where L is the labels of region in frame *t-1*, i is index of non-corresponded region in frame t.

The cost is calculated for all *(L, i)* pairs. Correspondence is established between the pair that gives the lowest cost, with the cost being less than a threshold. The all parameters of each region are updated using linear low pass filter prediction models as following equations.

$$Position : P_n = P_n + (t_n - t_{n-1}) * V \tag{8}$$

$$Velocity : V_n = \alpha * V_n + (1 - \alpha) * \frac{(P_n - P_{n-1})}{t_n - t_{n-1}} \tag{9}$$

$$Ratio : R_n = (1 - \alpha) * \frac{T}{B} + \alpha * R_n \tag{10}$$

$$Color : D_n = \alpha * D_{n-1} + (1 - \alpha) * D_n \tag{11}$$

Where, P is the center position of the bounding box, T is the total number of foreground pixels, B is the size of the bounding box surrounded foreground pixels, V is velocity of the bounding box, t is frame in time, α is a constant value between 0 and 1 depending on time.

The process on correspondence continues till no pairs are left or the minimum cost rises above the threshold. In other words, the correspondences have been found between all regions in frames *t-1* and t, or there might be regions in frame *t-1*, which have not been corresponded to in frame t due to exist from the scene or due to occlusion, or there may be regions in frame t, which have not been corresponded to regions in frame *t-1*, because they just entered the frame. The position plus predicted velocity of the region exit/enter from/to scene are easily used for determining to have exited/entered the scene. If this is not the case, then a check for occlusion is made. While an occluded is determined, all the regions in occluded have merged in a single region in frame t. Now we need to update the parameters of the occluded region. For occluded region same cost function is applied for tracking.

Figure 16. For the classification process on detected foreground regions, the dynamic structures of a group of person.

In addition, at the tracking process, the center of the detected foreground region in the sequences is stored in a trajectory map. This is shown in Figure 18 as white lines. In Figure 18, two people moves toward each other. When they are occluded, they turn around each other, then go back away. The thick and thin white lines separately represent the trajectories of each object. The stored data in the trajectory map is used to implement each foreground region motions in the scene. When any foreground object in tracking is hidden to any non-motion area (like passing at the behind of parked car), or temporarily occluded by other foreground object as they pass, the detected data of that object may not be obtained at the low level processing. At that or similar situations, high level implementation procedure is activated to estimate the possible position of that object using the previous tracked data obtained from trajectory. At the following frames, if the low level data about it is not obtained, its confidence is reduced. If the confidence of that object drops below a given threshold, it is considered lost, and is dropped from the tracking list stored in the trajectory map. High confidence objects (ones that have been tracked for a reasonable period of time) will persist for several frames, so if an object is momentarily occluded but then reappears, the foreground object tracker will reacquire it. An example experimental results on a test image sequence includes occlusion example between the person and a group of person is shown in Figure 19. In Figure 19-a, a person and a group of person are tracked in the image sequence. The person under tracking enters to the group of person, and another a small group contains two persons exits from the big group, as shown in Figure 19-b. Then second person enters to the scene, the first person is still in the big group, and the other small group is also tracked in the following frame sequence, Figure 19-c. Final frame in Figure 19-d, first and second persons occluded with the big group separately exit from the big group, and the small group is still in scene and tracked. The skeleton structures of the foreground regions are also illustrated in Figure 19, respectively.

When the tracking object is out of the scene, the algorithm based on foreground region detection may not more track that object. If it is desired more tracking, another tracking algorithm is activated. This algorithm uses an appearance model learned from previous frames. To activate this algorithm there is need to use a CCD camera mounted on a platform controlled pan and tilt motors with similar study in [35].

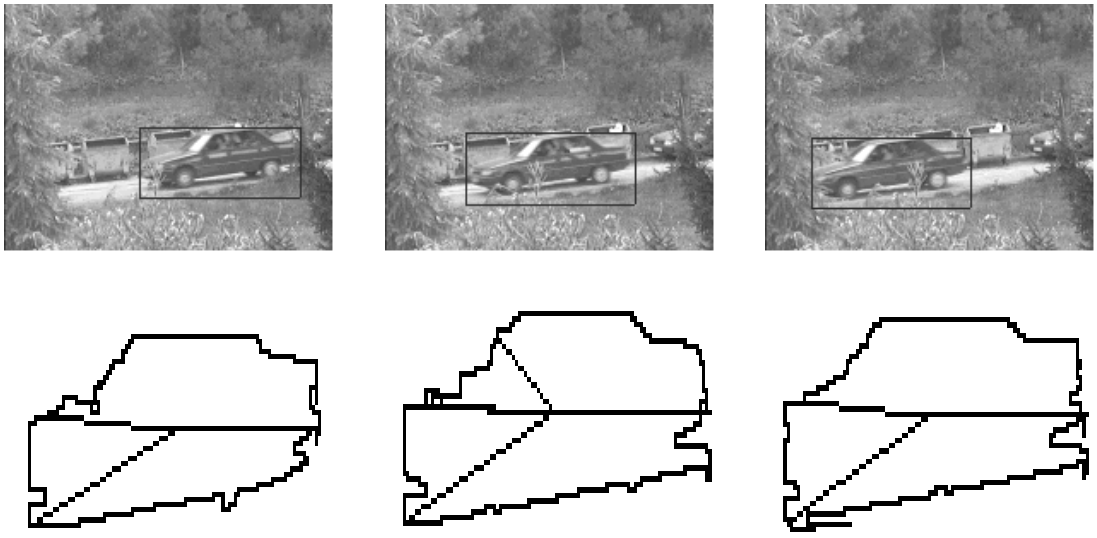


Figure 17. For the classification process on detected foreground regions, the dynamic structures of a car.

We have this system to develop full tracking process. It wasn't given more detail because this paper is not focused on that work.

4. Human Motion Analysis

Analyzing human motion for video applications is a complex problem. Real-world implementations will have to be computationally inexpensive and be applicable to real scenes in which objects are small and data is noisy. There has been considerable interest in the area of human motion tracking and analysis in recent years [2, 24, 12, 25, 26]. More references can be found in [28, 26]. For the human motion analysis, using the geometrical shape models has the advantage that they have much information than directly obtained features. But the difficulty and cost of calculation in extracting the models from the input frames are disadvantage of using shape models for real time video surveillance applications. Those difficulties prevent researches from concentrating on cognition part of motion analysis process. Consequently, an approach depends on the variations of the features produced from the silhouette motions in frames is presented in this study. The features implemented for human motion analysis are directly obtained from dynamic variations on the star skeleton structures of the silhouette shape. This basic idea behind of the human motion analysis presented is similar to the study in [22], and is an attempt to make motion analysis more robust for real time applications.

The key idea in this approach is that simple, fast extraction of the broad internal motion features of an object can be employed to analyze its motion. The star skeleton consist of the centroid of a motion blob and three local extremal points that are recovered when traversing the boundary (see section 2.3). The three local extremal points correspond to head, and two legs. A human is moving in an upright position, it can be assumed that the uppermost skeleton segment represents the torso, and the lower segments determined by two extremal points represent two legs. Then the angle θ measures between the upper-most extremal point and vertical, the angle α measures between the lower two extremal points, and the angle β also measures moving variations in time between end locations of two extremal points in 2D space corresponding to the

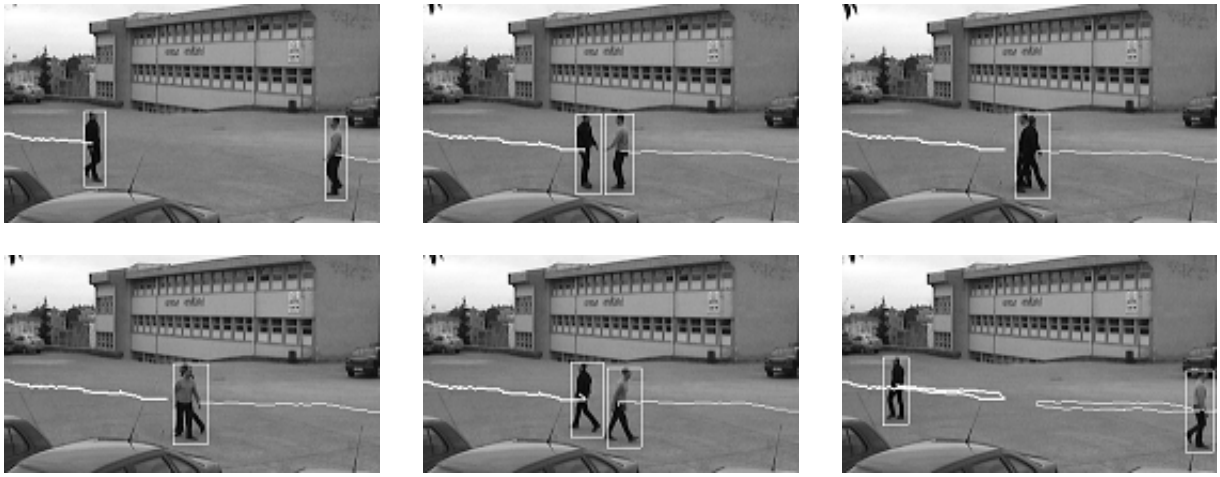


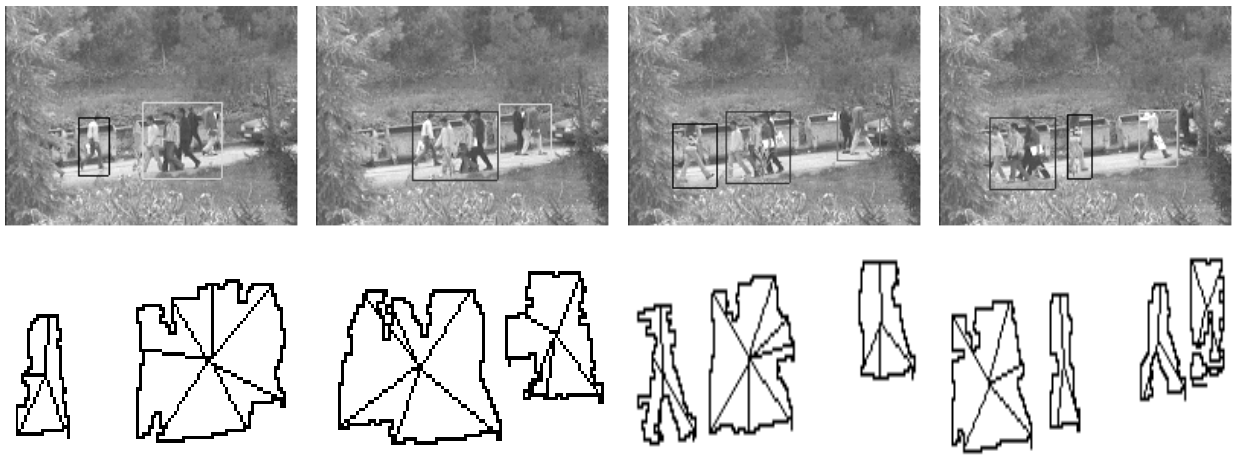
Figure 18. Tracking results on our data set (frames 2555-2606). Two person moving toward to each other, turn around each other, then go back away. Thick and thin white lines separately represent their trajectories produced by tracking algorithm.

ankles, as shown in Figure 20. (x_c, y_c) is the centroid of the motion blob (silhouette of the object under tracking).

In this section, an approach to distinguish the 'Walking' and 'Running' actions is presented as human motion analysis. The approach was developed and tested on the different test sequences in our database. Some sample frames in the test sequences are shown in Figure 21, where the black line with arrow represents the walking and the running paths.

The most important features involve the conditions used for distinguishing walking and running person can be produced by moving types of the foots in time, the characteristics on the foot cyclic and their speed variations [29]. That features can be easily and simply obtained by manipulating the star skeleton properties. Figures 22 a-b shows silhouette and skeleton motion sequences for walking and running person toward to the both directions. Figure 23 a-d plots the values θ_n, α_n, an acceleration of the centroid of the silhouette, and β_n over time. The actions in frames before frames numbered with 200 and after that are walking and running, respectively.

Examining the cyclic values of each angles shows that each angle has significant meaning for distinguishing two actions (walking, running), but the more robust human behavior understanding issue can be obtained by fusion the results of all that angles rather than implementing of the features alone. That is, the feature produced by the θ angle (represents posture of the person in action) can be manipulated to distinguish the running person from that of the walking person. But not all people lean forward when they run. In other words, there may be no big differencing between on some people lean forward when they run and walk. Figure 23-b plots θ angle variations in time on the skeleton motion sequence, some samples are shown in Figure 22, for a walking and then a running people, respectively. There is no big enough significant differencing between two sequences for the posture of the person in action, however, the β angle, as shown in Figure 23-d, has good enough significant and also presented an encouraged feature to distinguish walking-running actions in the test sequences. The producing of the reliable skeleton from the silhouette is important task because the β angle is directly obtained from both end points correspondence to the location of the ankles in the silhouette. Otherwise, the reliability of the implementation depends on the β angle might be possible reduced.



(a) Frame 809 (b) Frame 814 (c) Frame 821 (d) Frame 836

Figure 19. Occlusion example; a) A person and a group of person are tracked, b) The person under tracking enters to the group of person, and another a small group has two person exit from the big group, c) Then second person enter the scene, the first person is still in the big group, the other small group is also tracked, d) Finally, first and second person occluded with the big group separately exit from the big group, and the small group is still tracked.

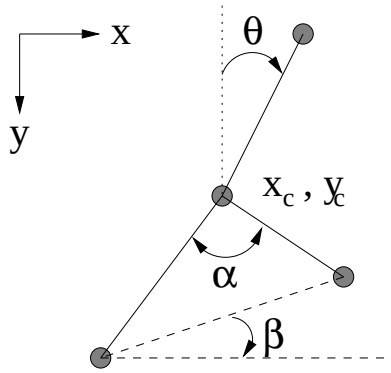


Figure 20. Determining of posture features from the skeleton: θ is the angle the torso makes with vertical, α is the angle between both legs, β is the angle between both ankle locations, and x_c and y_c are the spatial location of center of the skeleton which is corresponding to the hip position on horizontal and vertical directions, respectively.

When the variation in time on the α angle is considered, it is also giving one of good significant feature for distinguishing both actions. The acceleration on the silhouette in time is also producing the other good enough significant characteristics for analyzing both actions, as shown in Figure 23-c. Consequently, to be able to produce more robust and reliable human motion analysis, a fusion task which takes from each feature characteristics then fuses all the features together into a fused motion analysis using a weighted averaging process might be better [21]. This is one of the next studies for human motion analysis in the project presented. Implementation of the signals produced by each features of the skeleton is also another future work. Although it is not a main part of our work in this study presented, we have still given a beginner work on human motion analysis which is one of the part in the project, for completeness.

The producing of the reliable skeleton from the silhouette is important task because the important features obtained from the silhouette is directly obtained from different properties on the silhouette shape variations in time. The most of the studies in literature for human motion analysis basically depend on the silhouette of the human shape [1],[2],[33],[34]. As a result, the background estimation and upgrading

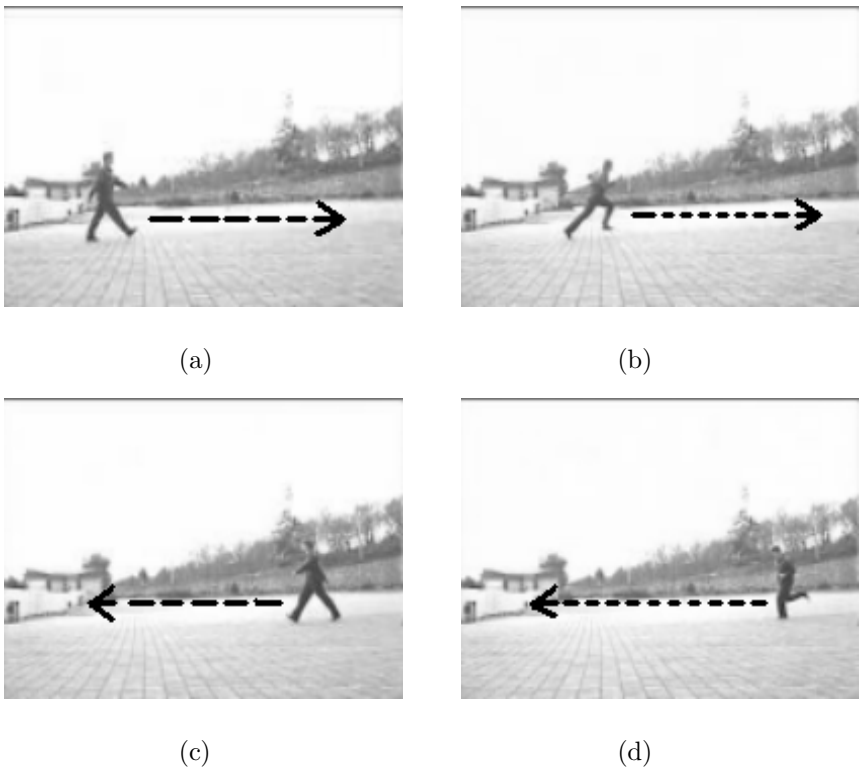


Figure 21. Some sample images in walking and running person toward different directions from the test motion sequences obtained from lateral view. (a) Walking (b) Running (c) Walking, (d) Running.

algorithms have naturally very critical main processing in the human motion analysis. This is one of the main part of our work presented, as explained in sections 2.1, 2.2. Shadow elimination is also another important and difficult task for more reliable silhouette unaffected from the shadow [12], [39], [4]. Test results for human motion analysis presented have been produced on the video sequences without having shadows in the scene.

5. Experimental Results

Extensive experiments are carried out to verify the effectiveness of the proposed algorithm. The following describes the details of the experimental results.

5.1. Data Acquisition

A video surveillance database is established for our experimental results. The database mainly contains video sequences on different days in outdoor and indoor environments. A digital camera (Sony DCR-TRV355E) fixed on a tripod and a CCD camera fixed on a pan-tilt motor platform are used to capture the video sequences. The database video sequences have mainly been classified as two main test sequences. The first type video sequences in two different indoor and in outdoor environments were used to test the algorithm approved for background estimation, maintenance, and object classification only. The other type video sequences on different days in outdoor environments were also used to produce experimental results on all the algorithm presented (background estimation, upgrading, classification, tracking and motion analysis).

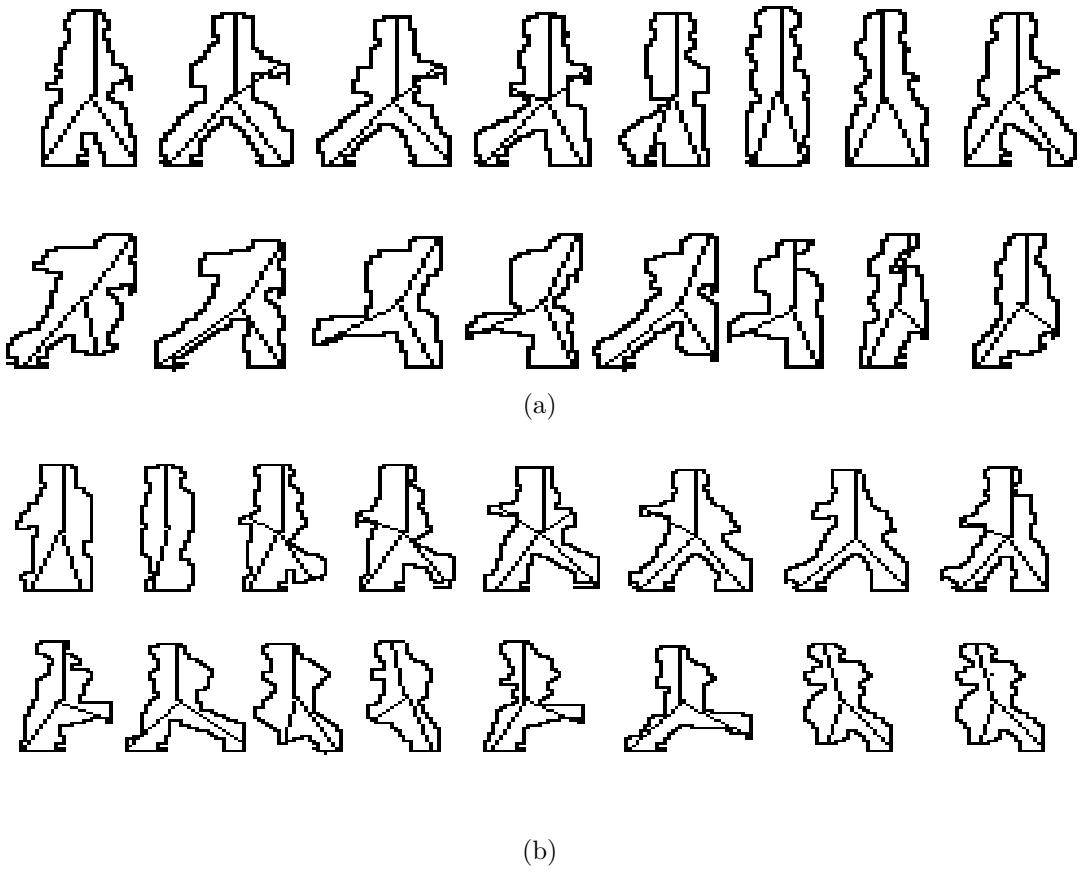


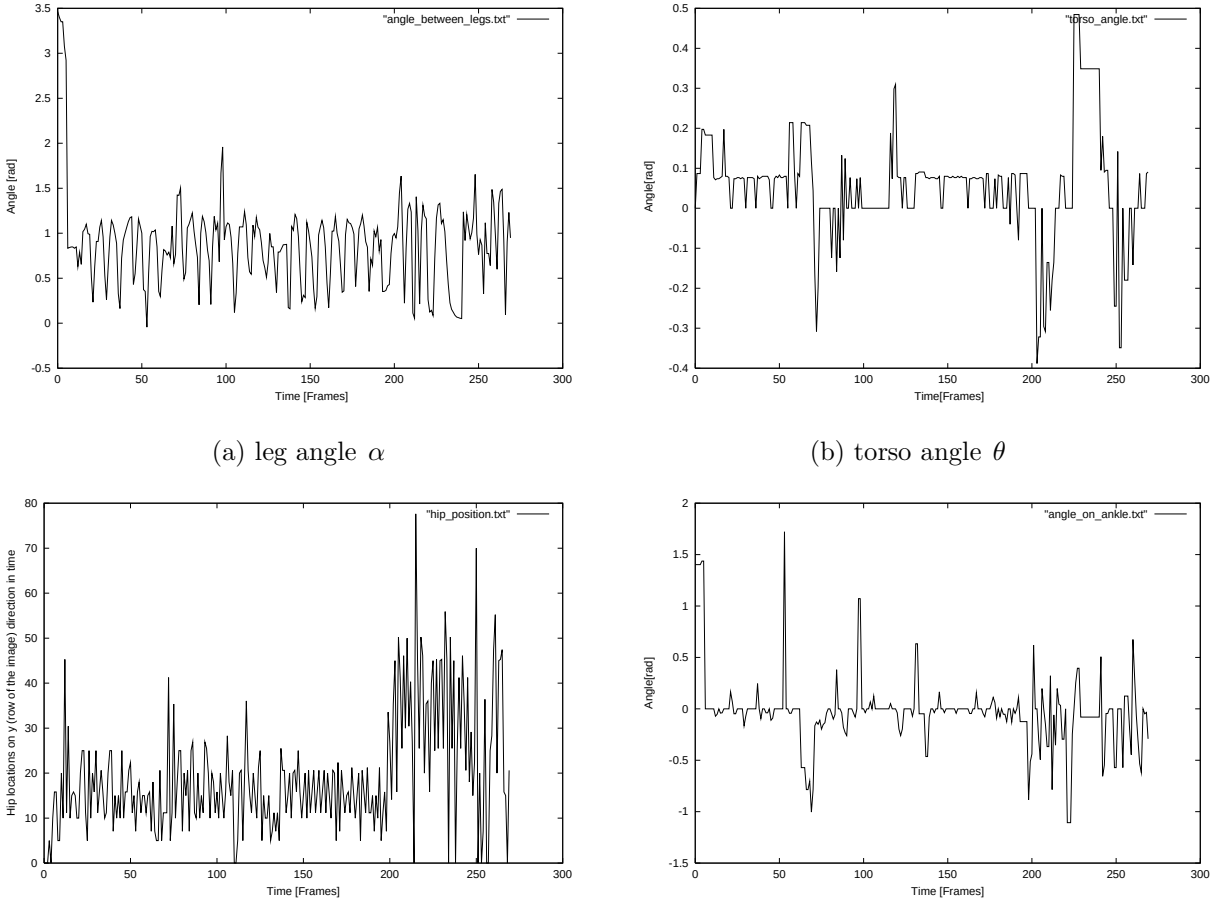
Figure 22. Silhouette and skeleton motion sequences of a walking and running person, respectively. (a) to the right way. (b) to the left way.

The following Table 2 summaries our database information used at the test processes of the algorithm presented. All images shown in this paper are some sample frames in our database. While the test sequences in our database were grabbed, the frame rate was 15.

5.2. Results

For each image sequence, we performed background estimation, maintenance, object classification as described in sections 2.1, 2.2, 2.3. The algorithm includes all previous steps and object tracking and motion analysis was tested on the image sequences numbered as outdoor 5 and outdoor 6 in the Table 2 as described in the sections 3, 4.

The algorithm for human motion detection, tracking and analysis presented in this paper has been implemented in C++ and runs under Windows 2K operating system at 96/133 MByte/MHz RAM, 850 MHz Celeron PC without using any special hardware. Currently, for 240 x 180 resolution gray-scale image sequences, the algorithm code without optimizing runs at 13-22 fps depending on the number of people in its field of view. This project was successfully demonstrated in a demonstration in ISCIS'2003 [19].



(a) leg angle α

(b) torso angle θ

(c) acceleration on the centroid point.

(d) ankle angle β

Figure 23. The periodic motion of α, θ, β angles, and acceleration on the centroid of the motion blob provide the cues to the person's activity. Walking activity is in frames before frame numbered with 200, at after the action is running activity detected on same person.

5.2.1. Results on Background Estimation

We compared our background estimation algorithm's output with that of several existing techniques used for background modeling. Tests were performed on several sequences representative of situations which might be commonly encountered in surveillance video. The HRR algorithm successfully learns, upgrades and models background scene to detect foreground objects, even when the background is not completely stationary (e.g. motion of tree branches) and distinguishes people from other objects (e.g. a car, as shown in Figures 15, 17) using shape and motion cues.

The reliability testing on the results for estimation background modeling by our algorithm and the comparison with the several algorithms for different examples are shown in Figure 2 and Figure 6, respectively. The some test results on the maintenance of the background model in real time is also shown in Figure 10. The algorithm presented for the background maintenance performs all processes in low level. This is one of the crucial advantages for real time video surveillance applications. The performance statistics of several algorithms and the algorithm approved are compared in Figures 8 and 9, on the important properties for real time applications. The properties are processing speed for the execution and requirement memory

Table 2. Overview of Our Test Database Used.

Environment	Objects	Motion Type	Sequence
Indoor 1	2 people	walking complex	1500
Indoor 1	2 people	walking complex	1500
Indoor 3	4 people	walking complex	2750
Outdoor 1	multi people, cars	walking complex, slow	2500
Outdoor 2	multi people, car	walking complex, slow	3250
Outdoor 3	5 people	walking complex in shadow	1500
Outdoor 4	5 people	walking complex, in shadow	1500
Outdoor 5	multi people, cars	walking, running complex	6500
Outdoor 6	single person	walking, running	1300

sizes for the implementation of the algorithm. From Figure 8, we can see that our background modeling algorithm presents very good encouraged results on requirement memory size for background estimation and maintenance. Once a more robust background model estimation is aimed, it is desired to process on the training sequence includes so much frames. This is typically 20-40 seconds (500-1000 frames) in [2], [5]. The length of the training sequence is one of the main reason for requirement so much memory sizes. However, one of the advantage of our algorithm is on this point, that is the basic idea of the algorithm is not depending on the length of the training sequence. The other more important advantage presented by our algorithm for real time applications on background model maintenance is the processing time as shown in Figure 9. Consequently, the HRR algorithm approved is simple but very effective method for background modeling and updating especially.

5.2.2. Results on Motion Analysis

The following experimental results are on the human identification, tracking, and motion analysis. With increasing demands of visual surveillance systems, human action analysis has recently gained more interest in the research studies. Human identification is not only depending on the skeleton of the silhouette but also the variations of the skeleton in a short time. Figures 15, 16, 17 show the the variations of the skeleton produced from silhouette in time for a person, a group of people, and a car. At the some frame, it is possible producing the similar skeleton structure for a person and a group of people. But an analyzing the variations of the skeleton structure in time with the other features (such as bounding box parameters) give more robust results on human identification.

For object tracking (a person and a group of person), after the object is detected, the tracking algorithm calculates the bounding box, the centroid and correspondence of each object over the frames. The tracking algorithm successfully handled occlusions between people. Entry of a group of people was detected as a single entry, however, as soon as one person separated from the group he was tracked separately as shown in Figure 19.

For human motion analysis, an approach similar in [22] but more reliable by adding more important features was presented. Normal Walking and Running actions in the surveillance scene were only considered to differentiate from each other. Four main parameters (θ, α, β, acceleration variations on the centroid of the motion blob, for more details see to section 4) are basically implemented to analysis two actions. In addition, the speed of the bounding box surrounding of the silhouette detected could be more considered for analyzing. But the basic approach presented will be developed to extrapolate for the future studies to analysis the other

possible people actions such as jumping, sitting, standing, lying, *etc.* For that reasons, the basic idea on the human motion analysis has been focused to the features of the silhouettes. The four parameters basically involve the features of the silhouette for two action (walking, running) analysis [29]. Test results shown in Figures 23a-d, encourage to implement this kind of parameters for human motion analysis for real time video surveillance applications. But the shadow is important problem for the silhouette based motion identification and analysis because the structure shape of the silhouette may be fluctuated by the shadow types. Test results produced for human motion detection and analysis were obtained in the surveillance area without having a shadow.

5.3. Discussions and Future Work

To provide a more general approach to human motion detection and analysis in unconstrained environments, much work remains to be done.

- The main strength of the algorithm for background estimation is that while the decision at each pixel is independent of its neighbors, it is not only based on past values observed at that pixel, but also local motion information. Additionally, by estimating the background independently at the pixel-level rather than at the region-level, the algorithm for background estimation reduces the amount of spatial clustering in the error, and therefore well suited for segmentation of moving objects.

- Although our results on human motion detection and analysis are encouraging, we are limited in our ability to extrapolate them. Test sequences used to produce results presented has not included any strong shadows in the scene. The structure of the silhouette may be big probably changed due to the shadow types. The next studies will be focused on this kind of problem to produce more robust motion analysis in different weather conditions. For that aim, we are planning to establish a larger database which will scan multiple day environments (strong sunny, rainy, snowy days), multiple views, clothing variations, *etc.*

- The main drawback of the human motion detection and analysis approach presented is that it is view-dependent, which is analogous to the state of the art of past algorithms [1], [2], except [31], [32]. The same feature extraction from the silhouette with different viewing angles has different recognition ability [33], [34]. Therefore, more reliable approach for real environments will be able to determine the sensitivity of the features to viewing angles. The later work for the view angle, an obvious way to generalize the view is to store training sequences taken from multiple view points, and classify both the subject and the view point [36].

- At the human motion detection and especially analysis, not only the approach for video surveillance applications depends on dynamic information but also the static information. So 3D human body modeling and tracking might prove to be of benefit [37], [38]. In a word, such combination of both informations will be developed in later work.

- In the future work, cyclic detection of the signals produced by the parameters on the silhouette will be developed for human motion analysis. It will be also extended to more complex human motion analysis such as jumping, crawling, and so on.

6. Conclusions

This paper has described an approach includes possible main steps in a real-time video surveillance system for human motion detecting and analysis for indoor and outdoor environments. It has operated on monocular gray-scale video imagery from a CCD camera. Firstly, an approach a simple but effective and robust for real time background estimation was presented. It learns and models background scene statistically to detect foreground objects, even when the background is not completely stationary (*e.g.* motion of people and/or tree branches) and distinguishes people from other objects (*e.g.* cars, animals.) using skeleton based shape and motion cues. The background model presented here was updated periodically to eliminate non-motion regions while at the beginning in motion, such as, a vehicle traveling in the scene and then parked. The background model was also upgraded every frame using temporal filtering to adapt small changes in the scene such as illumination changes (the sun being blocked by clouds causing changes in brightness). To classification object detected in motion, a skeleton process was developed in real time applications. Finally an approach for the human motion analysis depends on the silhouette was presented. Two actions, walking and running, have been successfully distinguished using four parameters produced from the silhouette tracked in time by the approach presented. The experimental results have presented that the proposed algorithm has an encouraging background based silhouette production and human motion analysis performance with relatively low requirement memory and computational cost.

Acknowledgment

This work is supported in part by the Science Foundation of Karadeniz Technical University, Turkiye (Grant No. 2002.112.009.1).

References

[1] R.T. Collins, A. J. Lipton, H. Fujiyoshi, T. Kanade, *Algorithms for Cooperative Multi sensor Surveillance*, Proceeding of IEEE, Vol. 89. No.10, 2001.

[2] I. Haritaoglu, D. Harwood, L.S. Davis, *W4: Real-Time Surveillance of People and Their Activities*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No.8, 2000.

[3] A. Bobick and J. Davis, *The Recognition of Human Movements Using Temporal Templates*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, No.3, March 2001.

[4] O. Javed and M. Shah, *Tracking and Object Classification for Automated Surveillance*, ECCV'2002, European Conference on Computer Vision, Copenhagen, Denmark, 2002.

[5] I. Haritaoglu, M. Flickner, *Detection and Tracking of Shopping Groups in Stores,* Proceeding of the 2001 IEEE Computer Vision and Pattern Recognition, Vol. 1, 8-14 December, 2001.

[6] P. Perez, C. Hue, J. Vermaak, M. Gangnet, *Color-Based Probabilistic Tracking,* Proc. of European Conference on Computer Vision, Copenhagen, 27 May- 2 June 2002, Denmark.

[7] Mubarak Shah, *Understanding human behavior from motion imagery,* Machine Vision and Applications, Special Issue: Human modeling, analysis, and synthesis, Vol. 14, Issue 4, pp. 210-214, September 2003.

[8] A. K. Jain, A. Ross, S. Prabhakar, *An Introduction to Biometric Recognition*, IEEE Transactions on Circuit and Sytems for Video Technology, Special Issue on Image- and Video-Based Biometrics, Vol. 14, No.1, pp. 4-20, January 2004.

[9] W. Grimson, C. Stauffer, R. Romano, L. Lee, *Using Adaptive Tracking to Classify and Monitor Activities in a Site*, in Proceeding of IEEE Conference on Computer Vision and Recognition, 1998.

[10] S.Ju, M. Black, Y. Yacoob, *Cardboard People: A Parametrized Model Of Articulated Image Motion*, International Conference on Face and Gesture Analysis, 1996.

[11] W. Long and Y. H. Yang, *Stationary Background Generation : An alternative to the Difference of Two Images*, Pattern Recognition, vol. 23, no. 12, 1990.

[12] C . Wren, A. Azarbayejani, T. Darrell, A. Petland, *Pfinder: Real-Time Tracking of the Human Body*, IEEE Trans. on Pattern Analysis and Machine Vision Intelligence, July 1997, Vol. 19, no. 7.

[13] D. Gutshess, M. Trajkovic, E. Cohen-Sola, D. Lyons, A. K. Jain , *A Background Model Initialization Algorithm for Video Surveillance*, IEEE Int. Conference on Computer Vision, 2001.

[14] K. Toyama, J. Krumn, B. Brumit, B. Meyers, *Wallflower: Principles and Practice of Background Maintenance*, 7th IEEE International. Conference on Computer Vision, November, 1999.

[15] A. Elgammal, D. Harwood, and L. Davis, *Non-parametric model for background subtraction*, in Proceeding 6th European Conference on Computer Vision, Dublin, Ireland, 2000.

[16] Y. H. Yang and M. D. Levine, *The Background Primal Sketch: An Approach for tracking moving objects*, Machine Vision and Applications, vol.5, 1992.

[17] C. Stauffer, and W. Grimson, *Learning Patterns of Activity using Real-Time Tracking*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No.8, August 2000.

[18] Y. Ricqueburg and P. Bouthemy, *The Recognition of Human Movements Using Temporal Templates*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No.8, August 2000.

[19] M. Ekinci, E. Gedikli, *Background Estimation Based People Detection and Tracking for Video Surveillance*, Springer LNCS 2869, ISCIS 2003, Computer and Information Sciences, 18*th* International Symposium, pp. 421-429, Turkey, November, 2003.

[20] P.L. Rosin, T. Ellis, *Image Difference Threshold Strategies and Shadow Detection*, in Proceeding. British Machine Vision Conference, 1995.

[21] M. Ekinci, F. W. Gibbs, B. T. Thomas, *Knowledge-Based Navigation for Autonomous Road Vehicles*, Turkish Journal of Electrical Engineering and Computer, Vol. 8, No. 1, 2000.

[22] H. Fujiyoshi, A. J. Lipton, *Real-time human motion analysis by image skeletonization*, Proceeding of the Workshop on Applications of Computer Vision, October, 1998.

[23] J. Vass, K. Palaniappan, X. Ahuang, *Automatic Spatio-Temporal Video Sequence Segmentation*, in Proceeding IEEE International Conference on Image Processing, 1998.

[24] S. Watcher, H. H. Nagel, *Tracking persons in monocular image sequences*, Computer Vision Image Understanding, Vol. 74, pp. 174-192, June 1999.

[25] R. Cutler, L. S. Davis, *Robust real-time periodic motion detection, analysis and applications* IEEE Trans. Pattern Analysis Machine Intelligence, Vol. 22, pp. 781-796, August 2000.

[26] L. Wang, W. Hu, T. Tan, *Recent developments in human motion analysis*, Pattern Recognition, Vol. 36, pp. 585-601, 2003.

[27] D. Gavrila, *The Visual Analysis of Human Movement: A Survey,* Computer Vision and Image Understanding, Vol. 73, No. 1, pp. 82-98, 1999.

[28] J.K. Aggarwal, Q. Cai, *Human Motion Analysis: A Review,* Computer Vision and Image Understanding, vol. 73, n. 3 pp. 428-440, March 1999.

[29] T. Mori, K. Tsujioka, T. Sato, *Human-like Action Recognition System on Whole Body Motion-captured File* Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Mani, Hawai, USA, Oct. 29 - Nov. 03, 2001.

[30] S.J. McKenna, et al., *Tracking groups of people,* Computer Vision and Image Understanding, vol. 80, (1) 2000, pp. 42-56.

[31] A. Johnson, A. Bobick, *A multi-view method for gait recognition using static body parameters,* in Proceeding of 3rd International Conference Audio and Video-Based Biometric Person Authentication, pp. 301-311, 2001.

[32] C. BenAbdelkader, R. Culter, L. Davis, *Stride and cadence as a biometric in automatic person identification and verification,* in Proceeding International Conference on Automatic Face and Gesture Recognition, pp. 372-376, 2002.

[33] L. Wang, T. Tan, H. Ning, W. Hu, *Silhouette Analysis-Based Gait Recognition for Human Identification,* IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 25, No. 12, December, 2003.

[34] L. Wang, T. Tan, W. Hu, H. Ning, *Automatic Gait Recognition Based on Statistical Shape Analysis,* IEEE Transactions on Image Processing, Vol. 12, No. 9, September 2003.

[35] R. Collins, O. Amidi, T. Kanade, *An Active Camera System for Acquiring Multi-view Video,* in Proceeding of international Conference on Image Processing, ICIP'02, 2002.

[36] R. Collins, R. Gross, J. Shi, *Silhouette-Based Human Identification from Body Shape and Gait,* in Proceeding of international Conference on Automatic Face and Gesture Recognition, pp. 366-371, 2002.

[37] S. Dockstader, K. Bergkessel, A. Tekalp, *Feature Extraction for the Analysis of Gait and Human Motion,* in Proceeding of International Conference on Pattern Recognition, pp. 5-8, 2002.

[38] B. Bhanu, J, Han, *Individual Recognition by Kinematics-Based Gait Analysis,* in Proceeding of International Conference on Pattern Recognition, pp. 343-346, 2002.

[39] T. Horprasert, D. Harword, L. Davis, *A Statistical Approach for Real Time Robust Background Subtraction and Shadow Detection,* IEEE Frame Rate Workshop, 1999.

[40] A. Jain, R. Bolle, S. Pankatti, *Biometrics: Personal Identification in Networked Society*, Kluwer Academic Publishers, 1999.