MOVING OBJECT DETECTION, TRACKING AND CLASSIFICATION USING NEURAL NETWORKS

By

HAMSA A. ABDULLAH

Assistance Lecturer, College of Information Engineering, AL-Nahrain University, Baghdad, Iraq I

ABSTRACT

Moving object Detection and Tracking (D&T) are important initial steps in object recognition, context analysis and indexing processes for visual surveillance systems. It is a big challenge for researchers to make a decision on which D&T algorithm is more suitable for which situation and/or environment and to determine how accurately object D&T (real-time or non-real-time) is made. There is a variety of object D&T algorithms (i.e. methods) and publications on their performance comparison and evaluation via performance metric. This paper discusses a hybrid technique for detecting and tracking moving pedestrians in a video sequence. The technique comprises two sub-systems: the first one for detecting and tracking moving objects in the visual field, and the second one for classifying the moving objects being tracked as human or cars by using MLP neural network. Experiments measuring the neural networks accuracy at classifying unseen computer generated and real moving objects are presented, along with potential applications of the technology. In this paper, the author developed robust routines for detecting and tracking moving objects with occlusion. The proposed model has proved to be robust in various environments (including indoor and outdoor scenes) and different types of background scenes. The experimental results prove the feasibility of the proposed method. Experiments on real scenes show that the algorithm is effective for object detection, tracking and classification.

Keywords: Image Processing, Object Detection, Object Tracking, Performance Metrics, Evaluation, Classification, Neural Network.

INTRODUCTION

Moving object detection is the basic step for further analysis of video. It handles segmentation of moving objects from stationary background objects. This not only creates a focus of attention for higher level processing but also decreases computation time considerably. Commonly used techniques for object detection are background subtraction, statistical models, temporal differencing and optical flow. Due to dynamic environmental conditions such as illumination changes, shadows and waving tree branches in the wind object segmentation is a difficult and significant problem that needs to be handled well for a robust visual surveillance system (Yi˘Githan, 2004).

Video object segmentation, detection and tracking processes are the basic, starting steps for more complex processes, such as video context analysis and multimedia indexing. Object tracking in videos can be defined as the process of segmenting an object of interest from a sequence of video scenes. This process should keep track of its motion, orientation, occlusion and etc. in order to extract useful context information, which will be used on higher-level processes. The context is defined as "any information that can be used to characterize the situation of an entity". Moreover, an entity can be a person, place or object that is considered to be related to users and applications. These applications are aimed for visual or non-visual information extraction and/or retrieval. When the camera is fixed and the number of targets is small, objects can easily be tracked using simple methods. Computer vision-based methods often provide the only non-invasive solution. Their applications can be divided into three different groups: Surveillance, control and analysis. The object Detection and/or Tracking (D&T) process is an essential item for surveillance applications. The control applications, which use some parameters to control motion estimation and etc., are used to control the relevant vision system. The analysis applications are often automatic, and used to optimize and/or diagnose system's performance. For well predefined (namely, annotated) datasets, the object recognition algorithms give good accuracy, rarely In the literature, the previous works concentrated mainly on moving-object D&T in videos. One can find bunch of methods dedicated to generic-object D&T in video processing like Background Subtraction (BS) , Mean-Shift (MS) and/or Continuously Adaptive Mean-Shift (CMS), Optical Flow (OF) , Active Contour Models (i.e. Snakes) and etc. Template matching is an essential-object D&T method, but it is simpler than others, and is generally based on matching a given template as an object in given a frame (Bahadır, 2010).

This paper describes a novel incorporation of an active contour model with a neural network categoriser. Combined, these systems provide a means of automatically tracking a moving object, and of determining whether or not that object is human.

Alternative methods exist for determining the class of an object being tracked, although these techniques generally rely either upon the object being centered in the image throughout the process, or upon complex models of the target object being formed on the fly, resulting in either an inability to support multiple objects in the same image, or large amounts of computation. The method we present involves training a neural network in advance, leaving very little computation to be performed while the object is being tracked. Figure 1 show the block diagram of moving object detection and classification system.

1. Literature Survey

Many researchers studied the field of Moving object detection, few of them focused on classification with neural networks.

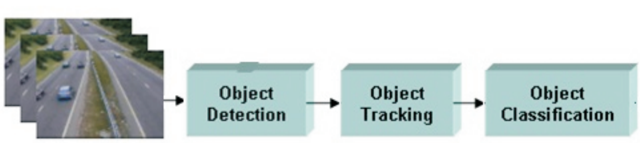


Figure 1. Moving Object Detecting and Classification System

In 2010 Sherin M. Youssef, Meer A. Hamza and Arige F. Fayed presented Detection and Tracking of Multiple Moving Objects with Occlusion in Smart Video Surveillance Systems. This paper present new method for detecting and tracking multiple moving objects based on discrete wavelet transform and identifying the moving objects by their color and spatial information. The proposed model has proved to be robust in various environments (including indoor and outdoor scenes) and different types of background scenes. The experimental results prove the feasibility of the proposed method. Experiments on real scenes show that the algorithm is effective for object detection and tracking (Sherin et al., 2010).

In 2011, Peter Dunne and Bogdan J. Matuszewski presented Histogram-based Detection of Moving Objects for Tracker Initialization in Surveillance Video. This paper presented an approach to localized object detection that is not dependent upon background image construction or object modeling. It is designed to work through camera embedded software using spare processing capacity in a visual signal processor. It uses a localized temporal difference change detector and a particle filter type likelihood to detect possible tractable objects, and to find a point within a detected object at which a particle filter tracker might be initialized (Peter,& Bogdan, 2011).

2. Object Detection

Each application that benefit from smart video processing has different needs, thus requires different treatment. However, they have something in common:

Moving objects detecting regions that correspond to moving objects such as people and vehicles in video is the first basic step of almost every vision system since it provides a focus of attention and simplifies the processing on subsequent analysis steps. Due to dynamic changes in natural scenes such as sudden illumination and weather changes, repetitive motions that cause clutter (tree leaves moving in blowing wind), motion detection is a difficult problem to process reliably (Yi˘Githan, 2004).

Moving object detection algorithms usually take two consecutive images as input and return the locations where differences are identified. These differences can be caused by the motion of an object, (including its entering and leaving the scene), changes in illumination or noise. The aim of such an algorithm is to locate only the changes that are due to structural changes in the scene, i.e. a moving object.

Moving object detection and extraction from the fixed background in the analyzed scene is mostly done by simple subtracting the current image and background image (that does not contain any moving objects).

The applied subtracting operation finds an absolute difference for each pixel, thus detecting moving objects (that have brighter or darker gray value), which usually differ from the background. If the difference is below a certain threshold, there is change in the scene and the observed pixel is regarded as if it belongs to the background.

Otherwise, there has been a change and the pixel belongs to the moving object. The absolute subtracting algorithm can be presented by:

$$\begin{array}{ll} IF \quad D = |C - B| > T & O = 1\ (Object) \\ ELSE & O = 0\ (Backgeound) \end{array}$$

Where C is the value of the corresponding pixel of the current image, B is the value of the corresponding pixel of the background image, D is the absolute difference of the current and background image and O is the binary difference image. T is the predefined threshold for image segmentation.

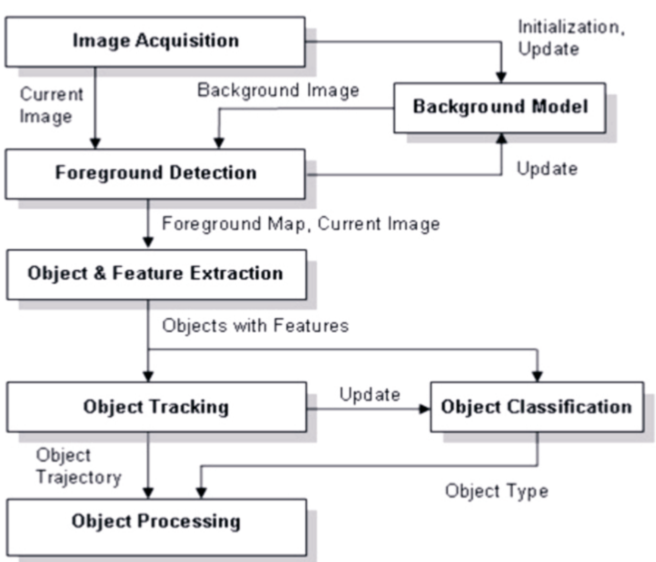


Figure 2. Proposed Method

In the case of fixed threshold it can happen that a moving object with an average brightness, which is only slightly different than the background, cannot be detected. The value for threshold becomes very important because:

- If the threshold is too low, a sudden increase in background brightness due, for example, to a rapid change from overcast to sunshine, could cause a false detection.
- If the threshold is too high, a moving object with brightness close to the background will not be detected.

The optimal threshold value is usually determined by analyzing the histogram of difference image in a certain time interval, where the appearance of moving object in the scene causes the histogram of difference image to widen. However, this is a time consuming process that is not effective in real-time applications. The main problem with difference technique is a variation in background brightness, mostly due to weather phenomena (clouds, rain, etc.) or artificial sources (illumination, car or plane headlights, shadows, etc.).

In order to make the background differencing technique more effective, the changes in ambient lighting must be compensated by some kind of background updating technique (Vesna et al. 2004). Figure 2 shows the proposed method for moving object detection and classification.

2.1 Background Updating

Every change of illumination in the analyzed image demands an adequate background updating. However, the interruption of image processing in real-time for background updating is not always possible. Thus, the background updating method must be simultaneous with image processing.

The simplest algorithm for background updating is the moving averaging updating technique, described in (Bahadır, 2010). as where Bn+1 is the updated background image, used for moving object extraction from the next image in the sequence, Bn is the previous background image and C_n is the current image, k is the constant that determines the updating rate. Typical values of k are ≈ 0.5 so that the influence of the current picture on background updating is equal to the influence of the previous background frame. The most important drawbacks of this method are the following: the moving object appears in the background image and the optimal choice of k is almost unsolvable problem.

Illumination changes in the scene are usually smaller than those due to the moving object in the analyzed image. This fact implies the possibility that the background updating is applied only to those segments of the analyzed scene that are not covered with the moving object. Pseudocode of this technique, applied on pixel level, is described by (Vesna et al. 2004).

IF ($D_n = |C_n - B_n| > T$)

$O_n = 0$ (Object)

$B_{n+1} = B_n$ (no background updating, there is a moving object in the scene)

ELSE

$O_n = 1$ (background)

$B_{n+1} = kB_n + (1\text{-}k)C_n$ (background updating , no moving objects in the scene)

This algorithm shows better results than the previous background updating algorithm. But the effectiveness and the quality of the selective background updating technique mostly depends on the proper choice of the threshold value T, which implies acompromise between background updating and moving object detection quality. If the threshold is not selected properly, the moving object pixels are misclassified as the background pixels. The background image becomes unusable.

Previously described techniques for background updating could be combined, giving selective background updating with averaging that provides new quality. Here, the background of the selected pixels is replaced by the average of the current and background picture pixels (k =0.5), instead of directly replacing the background pixels by the current image pixels.

We have implemented selective background technique with averaging only in those frames where the average value of variance of the ratio of background and current picture gave binary picture which number of pixels that indicate the moving object exceeded certain threshold.

2.2 Foreground Detection

Foreground detection compares the input video frame with the background model, and identifies candidate foreground pixels from the input frame. To obtain this classification, the difference map is usually binarized by thresholding. The correct value of the threshold depends on the scene, on the camera noise, and on the illumination conditions. In the following subsections we will discuss first how to generate the difference map given the background model and the current frame, and then we will discuss the thresholding techniques to obtain foreground-background classification (Shireen et al. 2008).

2.3 Pixel Level Post-Processing

Median filter is a commonly used image process, we need to reduce noise before image processing, median filter algorithms determine the principles of an odd pixel window W, window size of each pixel arranged according to Gray, middle gray value instead of the original F(i,j) the gray value, gray value as the center of the window g(i,j).

$g(i,j)=\text{median}\{F(i-k, j-l), (k, l \in W)\}$

Where W is the selected window size, F(i-k,j-l) for the window W of the pixel gray value, usually an odd number of pixels in the window.

2.4 Detecting Connected Regions

Morphological erosion and dilation in the image processing is an important foundation. Dilation is the image of the object's mathematical computing size, the dilate is a collection of the operations defined (Ssu-Wei et al., 2011).

$$A \oplus B = \{z \mid (\hat{B})_z \cap A \neq \emptyset\}$$

Where ϕ the empty set and B as structural elements. The erosion is the image of objects smaller or thinner, erosion and dilation similar to the mathematical definition.

$$A \ominus B = \{z \mid (B)_z \cap A^c \neq \emptyset\}$$

Erosion of A by B is a structural element of the origin of all the set positions, in which translation of the B and A's background does not overlap.

3. Object Tracking

The motion objects are tracked by managing the motion objects' feature vectors between consecutive frames. This section introduce the feature vector of a motion object defined in this paper, the object tracking method, and the elimination of the false motion detections. A feature-based object tracking algorithm requires useful feature selection, feature extraction, feature matching and proper handling of object's appearance and disappearance. An effective management of object entry and exit was proposed by Stauffer. Most of the works on tracking use a prediction on features in the next frame and compare the predicted value with estimated value to update the model. Usually a model like Kalman filter is used for prediction (Mir et al., 2006).

3.1 Background Modeling and Foreground Object Segmentation

To model the background, The author used a statistical method. The background image is constructed based on the statistical observation of pixel intensities of both the foreground and the background simultaneously. For every pixel, we developed a histogram of RGB color and considered the color with highest frequency. We used background subtraction for identifying regions where the objects are moving. We performed background subtraction in HSV color space, as HSV color space works well against shadow. We utilized the advantages of all the components of HSV color space to get more accurate result. We considered HSV color space for moving region segmentation; later we used RGB color space for feature calculation and histogram analysis. During background subtraction, finding a good threshold value is a major problem. If we take a smaller value for T to consider all the pixels in a moving region, then we introduce noise and shadow in the resultant image. If we increase T to remove shadow and noise, then we remove the self shadow region of the moving people and the image blob gets distorted. We can use different threshold value for different pixels and update them dynamically, instead of taking one global threshold. This approach is computationally expensive. To avoid these problems, we adopted K-means clustering technique for segmenting foreground pixels from background. In this approach, first the author calculate a difference matrix by subtracting background image from a frame. K-means clustering is then applied to the difference matrix to separate all the pixels into 2 clusters, a background cluster and a foreground cluster. This approach is highly efficient and eliminated the requirement of threshold. After finding the moving regions, the noise is removed by morphological operations (erosion and dilation).

3.2 Feature Extraction

3.2.1 Finding image blobs

The coherent pixels are grouped together as image blob by seeded region growing approach. The idea used in this approach is similar to seeded region growing, but different in terms of number of regions and choosing seeds. We try to grow one region at a time until all connected neighbouring pixels are considered and then start growing another region. After finding all image blobs, smaller ones are discard. The minimum size of blobs is determined by some heuristics and zoom of the camera. In our experiments, a minimum blob size of 200 to 300 pixels worked well.

3.2.2 Finding features of blobs

In this method, the author considered following significant features of blobs for matching during Euclidean distance-based approach and correlation-based approach:

- Size of blob
- Average of individual RGB components
- Coordinate of center of blob
- Motion vector

Size of the blob is represented as total number of pixels in the blob. The motion vector is calculated by taking the difference between coordinates of centers of blobs in two consecutive frames. Histogram of RGB color components was used during histogram-based matching. In the histogram, they considered a bin size of 10 and hence, there were a total of 26 bins for each color component.

The size of the bin was taken based on heuristics. A bin size of 1 is not computationally feasible. Moreover, they are very sensitive to slight variations of color. Taking a large bin size will work poorly during matching. As large sized blob have larger frequency count in histogram and vice versa, we normalized the values of the histogram within 1 by dividing the value by size of blob. All other features are also normalized to 1 before matching. For example, the size of the blob is divided by total size of images (240x320 in our case). Similarly the average color components are divided by 256 to normalize within 1.

The coordinates of center of blob are normalized by dividing each dimension of image.

3.3 Tracking object

The author developed the tracking system based on the basic tracking algorithm, which is as follows:

- Predict positions of known objects
- Associate predicted objects with current objects
- If tracks split, create new tracking hypothesis
- If tracks merge, merge tracking hypotheses
- Update object tracking models
- Reject false alarms

Most of the tracking system is built on the basis of this algorithm, and therefore use prediction of features in the next frame. It reduces the search space, but predicting features requires use of a predictor like Kalman filter. It requires significant computation time to built and update the model. In our system, we skipped the prediction of features to save computation time; rather we compared features obtained in the previous frame with features obtained in the current frame.

4. Object Classification

Classification techniques may be categorized in terms of two criteria. Firstly, they can be classified as supervised and unsupervised depending on the involvement of a training dataset. Supervised classification techniques require training data to be defined by the analyst in order to determine the characteristics of each category. The image is, thus, assigned to one of the categories using the extracted discriminating information. Problems of diagnosis, pattern recognition, identification, assignment and allocation are essentially supervised classification problems since in each case the aim is to classify object into one of a pre-specified set of classes. Unsupervised classification, on the other hand, searches for natural groups, called clusters, of objects present within the data by means of assessing the positions of the objects in the feature space. They are automated procedures and therefore require minimal user interaction. Another distinction among classification methods can be made by considering the underlying philosophy and assumptions of the techniques. By this, they can be classified into two groups: statistical classification and non-statistical classification. Statistical classification procedures employ purely statistical estimations to derive some rules from the data, which leads to some assumptions. The most common assumption of this kind is that the frequency distribution of the data is in Gaussian (or normal) form. However, non-statistical methods do not make any assumptions about the frequency distribution of the data used, and do not use the statistical estimates. The minimum distance and maximum likelihood classifiers can be given as examples of statistical classification methods, whilst the Artificial Neural Network approach, Support Vector Mechanics and knowledge-based methods can be given as examples to non-statistical classification methods (Kaushik, & Amit, 2011).

4.1 Unsupervised Classification

In some cases, information concerning the characteristics of individual classes is not available. In such circumstances, an unsupervised classification technique is used to identify a number of distinct or separable categories. In other words, an unsupervised method is used to determine the number of separable groups or clusters in an image for which there is no a priori or insufficient ground truth information available. Such unsupervised methods can be viewed as techniques of identifying natural groups, or structures, within data. While applying an unsupervised method, the analyst generally specifies only the number of groups to be discriminated, and the method generates the specified number of clusters, in feature space (Isabelle, & Andr´e, 2003), that correspond to separable features. Determination of the clusters is performed by estimating the distances between the data in feature space. Unsupervised classification techniques generally require user interaction in specifying the number of groups to be recognized and in labeling the correctly identified areas with the individual feature (or class) label. Owing to the minimal amount of user involvement, they are usually considered as automated procedures.

In addition, the assumption, forming the basis of the unsupervised approach, that the object belonging to a particular class will have similar features in feature space, and all classes are relatively distinct from each other in feature space is difficult to satisfy in practice. Consequently, the accuracy of the results obtained by unsupervised classification methods is limited.

Hannu Kauppinen et. al. proposes a non-segmenting object detection technique combined with a Self Organizing Map (SOM) based classifier and user interface. The purpose is to avoid the problems with adaptive detection techniques, and to provide an intuitive user interface for classification, helping in training material collection and labeling, and with a possibility of easily adjusting the class boundaries.

4.2 Supervised Classification

Supervised classification may be defined as the process of identifying unknown objects by using the information derived from the training data provided by the analyst. The result of the identification is the assignment of unknown object to pre-defined categories. The main difference between unsupervised and supervised classification approaches is that supervised classification requires training data as input. The training data is used to extract the properties of each individual class within the training data.

Supervised classification methods may be grouped into two general categories: statistical and non-statistical algorithms (Neural Network, Support Vector Mechanics). In the statistical supervised approach, the information required from the training data varies from one algorithm to another. For example, the maximum likelihood classifier requires the mean vector and variance-covariance matrix for each class. In contrast, supervised non-statistical models do not use any statistical information to identify unknown objects present in an image. Instead, they use all the training data available. This is the principal feature that makes supervised non-statistical models more powerful than their statistical counterparts. As a result, no assumption is made about the frequency distribution of the data in supervised non-statistical models. However, the effect of any incorrect definition of training is more considerable in these models than in the statistical models. This is due to the fact that supervised non-statistical models take every individual training data into consideration, whereas statistical models use only the overall properties of the data. For example, in the estimation of the mean, the effect of misidentified data is smoothed by averaging.

As mentioned earlier, supervised classification is performed in two stages (a) training and (b) classification. In the training stage, the analyst defines the regions that will be used to extract training data, from which statistical estimates of the data properties are computed. In the classification stage, every unknown feature in the test image is labeled in terms of its similarity to specified features. If object is not similar to any of the classes, then it can be allocated to an "unknown" class. The characteristics of the training data selected by the analyst are of considerable importance for the reliability and the performance of a supervised classification process. The training data must be defined by the analyst in such a way that they accurately represent the characteristics of each individual feature used in the analysis.

Two features of the training data are of key importance (Stéphane et al., 2006). These are the representativeness (or objectiveness) and the size of the training data. In order to have a representative set of data, the sample selection must be performed by considering the different objects of various sizes at various positions and at various orientation of the object that correctly represent the diversity of each class, so that variations of object position and rotation are considered. The size of the training dataset is also very important if statistical estimates are to be reasonable.

Sample size is mainly related to the number of features whose statistical properties are to be estimated. Although supervised classification methods require more user interaction, especially in the collection of training data, they generally give more accurate results compared to unsupervised classification techniques. Therefore, researchers mostly favor them.

A new mathematical model that has emerged recently, and which has made a great impact in the scientific community is the Artificial Neural Networks (ANNs) (Justin, & Robert, 1994). ANN has attracted increasing attention from researchers in many fields during the last decade, resulting in studies aiming to solve a wide range of problems. ANN has been proved to be more robust compared to conventional statistical classifiers in recognizing patterns from noisy and complex data and in estimating their nonlinear relationships. In short, it is known to be good at learning the internal representation of data in any form.

In order to classify shapes as human or non-human using a supervised neural network as shown in Figure 3, it was necessary to obtain training and test set of examples and counter examples.

Each shape used as a training or test pattern had a snake locked onto it, which was then translated into an axis crossover vector. Axis crossover vectors containing 4, 8, 12, 16, 20 and 24 axes were used, each on 9 identical neural networks containing the corresponding number of input units. Each of the 9 identical neural networks were initialized with a different random weight matrix, to lessen the chances of a network becoming trapped in local minima in the weight space.

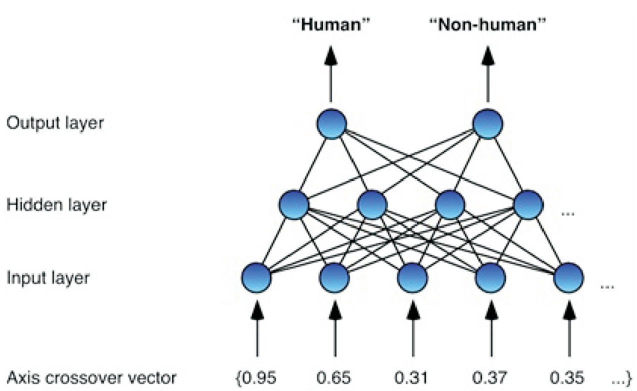


Figure 3. Neural Network

Shapes which the network is uncertain at classifying can be incorporated into the training set, so that the training set becomes more representative of the possible shapes which will be encountered, and improves the network's generalization abilities with these types of shape.

Of these 9 input unit networks, those containing 19 hidden units were best at distinguishing human from non-human shapes, being able to classify 90% of unseen human shapes and 100% of unseen non-human shapes correctly

To test the chosen network's confidence in its categorizations, it was necessary to look at the average difference between the values output by both of its output units during the previous experiment to see how 'confident' it was that a pedestrian vector was pedestrian and that a non-pedestrian vector was not.

4.3 Features

The researcher use temporal differencing to detect moving objects and for each detected region, we compute feature vectors for object type classification. To classify objects in a video stream, it is important to use a classification metric which is computationally inexpensive, reasonably effective for small numbers of pixels on object, and invariant to lighting conditions or viewpoint.

4.3.1 Shape-based Feature

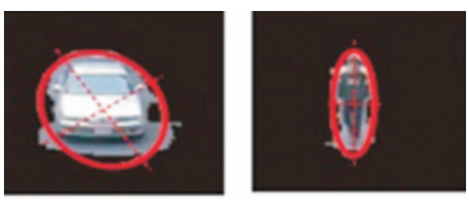
Aspect ratio Aspect ratio is determined relative size to measure two extensions of the object. We apply ellipse fitting onto detected regions as shown in Figure 4. The author apply the method to the task of distinguishing whether an image blob is a vehicle (hereafter referred as VH), a human (SH), and we determine that shape-based feature is reliable for this classification task (Masamitsu, & Hironobu , 2006).

$$Aspect\ Ratio = \frac{Length\ of\ Minor\ Axis}{Length\ of\ Major\ Axis}$$

5. Experimental Results

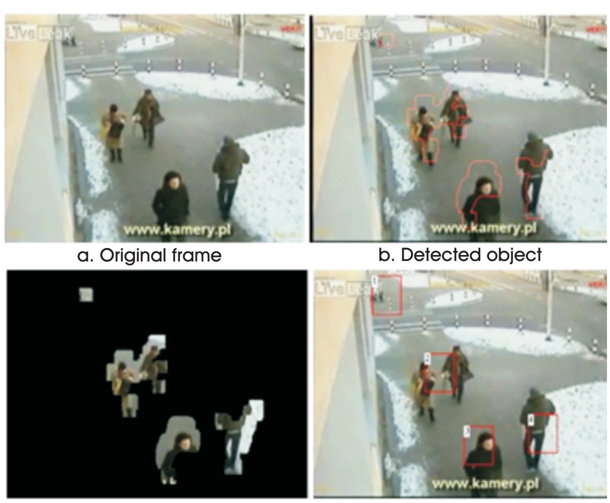
To verify the robustness and efficiency of the proposed algorithm, it has been applied for several video clips under dynamic environment, which includes various situations like partially or completely occlusion, disappearance, reappearance. The algorithm is written in MATLAB. Figure 5 shows the results of the proposed. In the left picture, a number of particles are distributed, then move around and eventually converge. The new tracker is showed in the right picture.

The author tested the method on various sequences including cars (Figure 5, pedestrians, Figure 6 interfering objects, figure 7 Cars, figure 8 interfering between human and cars, and figure 9 cars in night.). All sequences shown in Figures (5-9) are recorded on outdoor-scenes that include the sky, trees, buildings, grounds, and snow. They include several kinds of noise caused by illumination changes, small movement in the background, and reflection. However, our results showed remarkable robustness against these environments. This method succeeded detecting and tracking moving objects accurately in all video sequences in Figures (5-9), even though these sequences had many causes of noise. For instance, Figure 5 include ground covered by snow which causes reflection. The authors also succeeded in tracking



(a) VH (b) SH

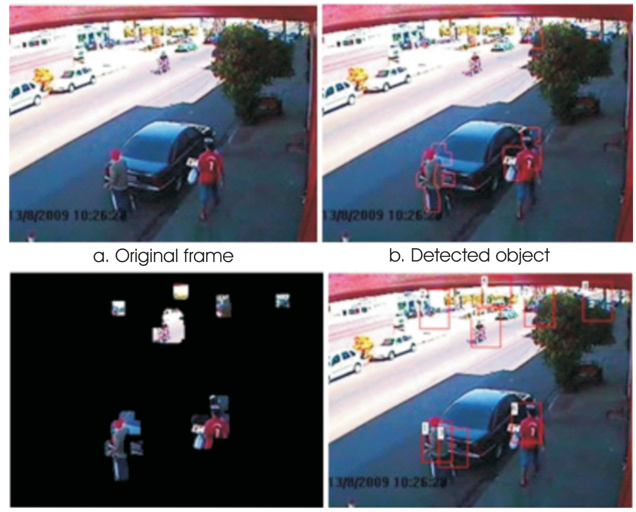
Figure 4. Shape-based features



a. Original frame b. Detected object

c. Deference frame d. Tracked object

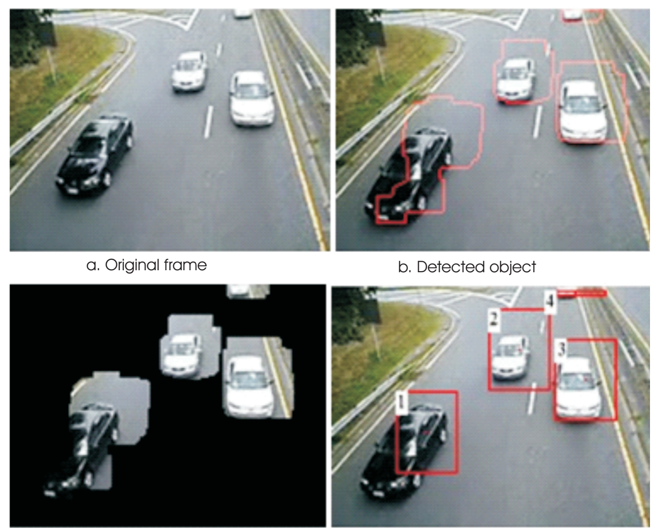
Figure 5. Pedestrians with Snow



a. Original frame b. Detected object

c. Deference frame d. Tracked object

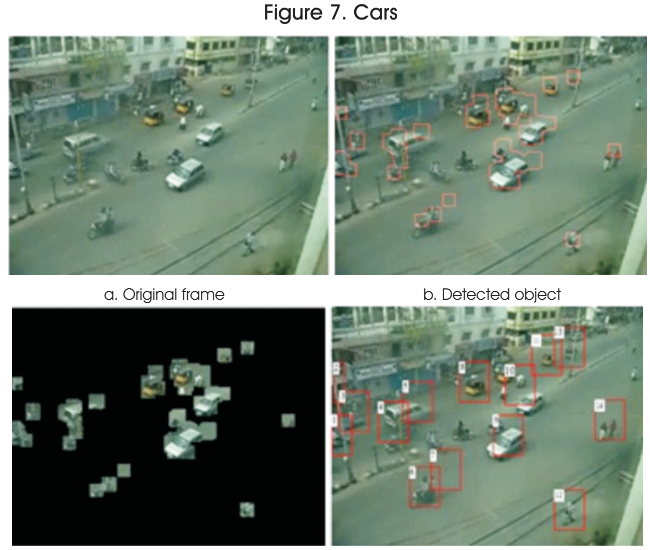
Figure 6. Pedestrians Interfering Objects



a. Original frame b. Detected object

c. Deference frame d. Tracked object

Figure 7. Cars



a. Original frame b. Detected object

c. Deference frame d. Tracked object

Figure 8. Interfering Human and cars

occluded cars and interfering pedestrians in Figures. 6,7,8,9 respectively. Table 1 shows the results of the classification of these objects. In classification the objects into two grope, the first grope is human and the second grope is cars. And the calcification divides the result into two face the testing face and training face. The results classify 90% of the objects.

Conclusion

In this paper, the author propose a modified method of motion detecting and object tracking. The proposed method can reduce noise, detect motions, track objects, and delete the light change and the shadows. It has been found that the classification of objects can be achieved by use of representative datasets and employing more powerful classification techniques. Artificial neural network is chosen as it is non parametric in nature. The proposed object tracking algorithm successfully tracks objects in consecutive frames. The tests in sample applications show that using nearest neighbor matching scheme gives promising results and no complicated methods are necessary for whole-body tracking of objects. The occlusion handling algorithm would fail in distinguishing occluding objects if they are of the same size and color. The methods the author presented for visual surveillance show promising results and can be both used as part of a real-time surveillance system or utilized as a base for more advanced research such as activity analysis in video.

Figure 9. Cars in Night

Training		Testing	
input	output	input	output
Car1	cars	Car6	cars
Car2	cars	Car7	cars
Car3	cars	Car8	cars
Car4	cars	Car9	cars
Car5	cars	Car10	cars
Human1	human	Human6	human
Human2	human	Human7	human
Human3	human	Human8	cars
Human4	human	Human9	human
Human5	human	Human10	human

Table 1. Classification Phase

References

[1]. **Bahadır, K. (2010).** Review And Evaluation Of Well-Known Methods For Moving Object Detection And Tracking In Videos. *Journal of Aeronautics And Space Technologies*, 4, 11-22. http://www.hho.edu.tr/HutenDergi/2010Temmuz/5-KARASULU.pdf.

[2]. **Isabelle, G., & Andr´e, E. (2003).** An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*,3,1157-1182. doi:10.1162/153244303322753616.

[3]. **Justin, D.P., & Robert, A.S. (1994).** Comparisons of Neural Networks to Standard Techniques for Image Classification and Correlation. *Proceedings, 14th Annual International Geoscience and Remote Sensing Symposium*, 3, 1404-1406. doi: 10.1109/IGARSS.1994.399452.

[4]. **Kaushik, A., & Amit, K. (2011).** Feature Extraction and Classification Technique in Neural Network. *International Journal of Computer Applications*, 35, 29-35.doi:10.5120/4383-6070.

[5]. **Masamitsu, T., & Hironobu, F. (2006).** Evaluating Feature Importance for Object Classification in Visual Surveillance. *ICPR '06 Proceedings of the 18th International Conference on Pattern Recognition*, 2, 978-981. doi:10.1109/ICPR.2006.510.

[6]. **Mir, A.A., S. Indupalli, & Boubakeur, B. (2006).** Tracking Multiple People for Video Surveillance. *First International Workshop on Video Processing for Security.* http://computer-vision.org/4security/pdf/windsor.pdf.

[7]. **Peter, D., & Bogdan J.M. (2011).** Histogram-based Detection of Moving Objects for Tracker Initialization in

Surveillance Video. *International Journal of Grid and Distributed Computing*, 4, 71-78. http://earticle.net/article.aspx?sn=153589.

[8]. **Sherin, M.Y., Meer, A.H., & Arige, F.F. (2010).** Detection and Tracking of Multiple Moving Objects with Occlusion in Smart Video Surveillance Systems. *5th IEEE International Conference on Intelligent Systems (IS),* 120-125. doi:10.1109/IS.2010.5548407.

[9]. **Shireen, Y.E., Khaled, M.E., & Sumaya, H.A. (2008).** Moving Object Detection in Spatial Domain using Background Removal Techniques - State-of-Art. Recent Patents on Computer Science,1, 32-54. http://www.benthamscience.com/cseng/samples/cseng1-1/Elhabian.pdf.

[10]. **Ssu-Wei, C., Luke, K.W., & Jen-Hong, L. (2011).** Moving Object tracking Based on Background Subtraction Combined Temporal Difference. *International Conference on Emerging Trends in Computer and Image Processing* (ICETCIP'2011) Bangkok Dec, 16-19. http://psrcentre.org/images/extraimages/1211649.pdf.

[11]. **Stéphane, A., Michael, B., & Vallipuram, M. (2006).** Off-line Signature Verification using the Enhanced Modified Direction Feature and Neural-based Classification. *IJCNN '06. International Joint Conference*, 684-691. doi:10.1109/IJCNN.2006.246750

[12]. **Vesna, Z., Željen, T., & Vojin, Š. (2004).** New Algorithm For Moving Object Detection. *Journal of Operations Research*, 1, 117-132. doi:10.2298/YJOR0401117Z

[13]. **Yi˘Githan, D. (2004).** *Moving Object Detection, Tracking and Classification for Smart Video Surveillance.* Ph.D. Thesis. http://signal.ee.bilkent.edu.tr/Theses/Yigithan.pdf.

ABOUT THE AUTHOR

Hamsa A. Abdullah is currently working as an Assistance Lecturer at College of Information Engineering (since 2008). She received M.Sc degree in Information Engineering from Information Engineering College, AL-Nahrain University, Baghdad, Iraq in 2008. Her research interests include Multimedia, Pattern recognition and Network Security.