Application and Implementation of Improve Image Processing and Feature Recognition Applied to Full-Field Measurements

Md. Amir Baig[1] Monika Tiwari[2]
[1.2]Assistant Professor, Electronics and Communication
[1.2]Jayoti Vidhyapith Women's University Jaipur

Abstract--- The understanding of crowd behavior in semi-confined spaces, it is important to design a new pedestrian facility, for major layout modifications in daily management sites that are subjected to crowd traffic. Conventional manual measurement techniques are not suitable for data collection of patterns of site occupation and movement. Real-time monitoring is tedious and tiring, but safety-critical. Also sometimes the probability of this measure will yield a false alarm and efficient methods for estimating this probability at run time. This paper presents some image processing techniques which, using existing closed-circuit television systems can support both data collection and on-line monitoring of crowds. The paper describes techniques to perform efficient and accurate crowd recognition in difficult domains. In order to accurately model small, irregularly shaped targets; the crowd objects and image are represented by their edge maps, with a local orientation associated with each edge pixel. Three-dimensional objects are modeled by a set of two-dimensional (2-D) views of the object by Translation, rotation, and scaling of the full three-dimensional (3-D) motion of the object. And this information can be used to maintain a low false alarm rate or to rank competing hypotheses based on their likelihood of being a false alarm. The application of these methods could lead to a better understanding of crowd behavior, improved design of the built environment and increased pedestrian safety.

I. INTRODUCTION

Human observers of crowds, particularly those experienced in the management of crowds in public places, can detect many crowd features, in some cases quite easily. Normally they can distinguish between a moving and a stationary crowd and estimate the majority direction and speed of movement of a large crowd, without needing to visually identify or count the separate individuals forming the crowd. They could also easily estimate in a qualitative way the crowd density.

Fig. 1: Edge Detection (Generalized case)

An idea is to measure the total perimeter of all the regions occupied by people. For low-density crowds, this can be expected to give a measure of density, although errors are inevitable as numbers increase because of occlusion and overlapping of individuals.

Fig 2: An image where pedestrian "edges" have been extracted and thinned. The number of remaining "picture elements" has been found to be correlated to the number of people in the image.

Edge-detection is a standard low-level image processing function which can be used to derive outlines of individuals and groups in a video image. The process can be refined further by thinning the edge images to minimize the effects of varying edge thickness.

Fig. 1 and fig. 2 shows a typical "thinned edges" image.

This paper considers methods to perform crowd recognition by representing target models and images as sets of oriented edge pixels and performing matching in this domain. While the use of edge maps implies matching 2-D models to the image, 3-D objects can be recognized by representing each object as a set of 2-D views of the object. Explicitly modeling translation, rotation in the plane, and scaling of the object (i.e. similarity transformations), combined with

All rights reserved by www.ijsrd.com

considering the appearance of an object from the possible viewing directions, approximates the full, six-dimensional (6- D), transformation space.

To require image processing and computer vision techniques to match such capabilities of human observers is at present unrealistic. However, study of the methods used by human observers may help in the choice of image processing algorithms likely to be useful in automatic assessment of crowd behavior.

A. Detection of crowd

1) Detection of Stationary Crowds

It is well-established that crowd congestion which is reaching a danger-level can be spotted by observers noting that the up-and-down oscillatory head movements of individuals walking in a freely-flowing crowd stop when the crowd is too dense for free movement. While Hentschel [1] investigated frequency-domain techniques to identify these up and-down movements to discriminate between stationary and non-stationary flow. A possible alternative is to compute the 2-dimensional Discrete Fourier Transform (DFT) for each image in a time sequence followed by a measurement of temporal changes in the resulting magnitude and/or phase spectra (frequency domain). This approach has two main disadvantages. First, the DFT for a single image is related to local changes of intensity and not to temporal (interframe) properties. Secondly, it involves a high computational and memory cost. A more effective method is to isolate motion properties in the image sequence through a data-reducing coding mechanism, such as the Discrete Cosine Transform (DCT) whose form for a one-dimensional "image" f (x, t) of N elements is given by:

$$g(t) = \sum_{x=0}^{N-1} f(x,t)\cos(2\pi kx), \quad \ldots..1$$

where k is a constant derived from the maximum signal frequency (Nyquist criterion) and the maximum expected motion to be observed. This transform associates sinusoids with the time-varying parts of an image (faster moving objects are associated with high frequency sinusoids and non-moving objects are associated with constant levels), which can be detected by applying the DFT on g(t) . For one-dimensional "images" of N pixels, the DCT calculates a single value for each image in a time-sequence (a data reduction of N to 1), therefore reducing significantly the computational cost of the DFT. As we are only concerned here with detecting movement in the vertical direction, the DCT of a two-dimensional image sequence is given by:

$$g(t) = \sum_{y=0}^{N-1}\sum_{x=0}^{N-1} \mathbf{I}(x,y,t)\cos(2\pi ky) \quad \ldots.2$$

where **I**(*x*, *y*, *t*) represents each image (of size NxN) in the sequence. Hentschel [2] developed and tested a number of variants of this scheme, finally proposing the "Linear Area Transform" (LAT) algorithm that takes an image sequence **I**(*x*, *y*,0) , **I**(*x*, *y*, Δ*T*) , **I**(*x*, *y*,2Δ*T*) , ... **I**(*x*, *y*,*m*Δ*T*) and first computes and *interframe* sequence **D**(*x*, *y*,0) , **D**(*x*, *y*, Δ*T*) , **D**(*x*, *y*,2Δ*T*) , ..., **D**(*x*, *y*, (*m* −1) Δ*T*) , by pixel-to-pixel subtraction between adjacent frames i.e. **D**(*x*, *y*,0) =|**I**(*x*, *y*,Δ*T*) −Δ**I**(*x*, *y*,0)| , etc. Non-zero pixels in the interframe sequence thus correspond to areas of movement in the images. The LAT then computes the scalar time sequence:

$$g(t) = \sum_{y=0}^{N-1}\sum_{x=0}^{N-1} \mathbf{D}(x,y,t) \quad \ldots.3$$

Where NxN is the image size. In other words, the total amount of motion in each image, weighted by vertical position, is accumulated in g(t) . To improve peak detection, the d.c. component of g(t) is removed by computing its derivative (difference) to which the DFT is applied. The frequencies of head oscillations correspond to peaks in the resulting frequency spectrum.

Fig. 3

Fig. 3: shows a typical result for a sequence with 32 images, where the peak at about 2Hz agrees with manual observations.

B. Estimation of Crowd Density

Human observers could be expected to estimate crowd density by counting individuals, making use of their ability to rapidly identify the separate individuals using higher-order knowledge about the shape and characteristics of humans. For example the heads of people in a crowd are easy for a human observer to locate, and so counting heads would be a natural approach.

image under analysis.

Fig. 4: A "background-only" images for the site in Fig. 3.

All rights reserved by www.ijsrd.com

This is used to isolate pedestrians from images by removing the surrounding background To expect an image processing and computer vision system to identify each individual in a crowd as a preliminary step to counting and tracking is unrealistic at present. Although many sophisticated systems have been devised for military applications (e.g. identifying and tracking multiple targets and decoys) and for some civil applications (vehicle tracking and identification on motorways, pig identification, etc.) there is not yet any realistic possibility of using such methods economically for crowd-behavior studies. In any case, the main theme of our research is to find general models of crowd behavior which do not rely upon detecting the behavior of individuals.

Fig. 5: An image where the background has been removed. The number of remaining "picture elements" has been found to be correlated to the number of people in the image.

It is clear that a human observer can easily distinguish a very dense crowd from the background (surrounding buildings, road surfaces, and so on) and would be likely to use the ratio of 'crowd area' to 'background area' as a rough estimate for the crowd density. This idea could be applied quantitatively to computer-based density estimation if the image-pixels corresponding to the crowd could be separated from those of the background. This can be done quite effectively using a 'reference image' of the scene, obtained with no crowd present (Fig. 4) for subtraction from the
A typical result is shown in Fig. 5. Care has to be taken that lighting conditions are similar, and that there are not other movable objects in the scene (such as vehicles, temporary bill boards or other signs, etc.) which the computer would not distinguish from people.

C. Estimation of Crowd Motion

Human observers have a highly developed capability for visual tracking of moving objects in a complex scene, which is not easily matched by computers. In semi-confined spaces individuals are free to move in various directions. Moreover, at any one instant, different parts of a single individual (e.g. head, limbs) move in different ways.
Recognizing movement in a sequence of video images, without identifying objects or "understanding" the scene, requires a technique which tracks the displacement (and hence the velocity) of regions of similar brightness-patterns from one image to the next. Of course, such velocity estimation cannot distinguish between individuals or separate the movement of humans from other moving objects. However, crowd analysis is usually more concerned with group behaviour such as preferential motion direction and magnitude. For instance, a typical useful measure is the distribution of the proportion of people moving in a discrete set of preferential directions (e.g. architects normally use a "wind rose" of eight directions: North, North East, etc.). The approach proposed here is therefore to measure motion features at pixel or pixel neighbourhood level which are then aggregated to obtain motion properties for larger regions in an image.
The aggregated results can then be used to establish overall preferential crowd velocities (direction and magnitude).

D. CROWD RECOGNITION

E. Matching Oriented Edge Pixels

This reviews the definition of the Hausdorff measure and how a generalization of this measure can be used to decide which object model positions are good matches to an image. This generalization of the Hausdorff measure yields a method for comparing edge maps that is robust to object occlusion, image noise, and clutter. A further generalization of the Hausdorff measure that can be applied to sets of oriented points is then described.

F. Hausdorff measure

The directed Hausdorff measure from M to I, where M and I are point sets, is

$$h(M,I) = \max_{m \in M} \min_{i \in I} \|m - i\| \quad \ldots.4$$

This yields the maximum distance of a point in set M from its nearest point in set I.
In the context of recognition, the Hausdorff measure is used to determine the quality of a match between an object model and an image. If M is the set of (transformed) object model pixels and I is the set of image edge pixels, the directed Hausdorff measure determines the distance of the worst matching object pixel to its closest image pixel. Of course, due to occlusion, it cannot be assumed that each object pixel appears in the image. The partial Hausdorff measure [2] between these sets is thus often used. It is given by

$$h_K(M,I) = K^{th}_{m \in M} \min_{i \in I} \|m - i\|. \quad \ldots..5$$

This determines the Hausdorff measure among the K object pixels that are closest to image pixels. K can be set to the minimum number of object pixels that are expected to be found in the image if the object model is present or K can be set such that the probability of a false alarm occurring is small. Since this measure does not require that all of the pixels in the object model match the image closely, it is robust to partial occlusion. Furthermore, noise can be withstood by accepting models for which this measure is nonzero, and this measure is robust to clutter that may appear in the image since it measures only the quality of the match from the model to the image and not vice versa.

G. The Generalization to Oriented Points

The Hausdorff measure can be generalized to incorporate oriented pixels by considering each edge pixel in both the object model and the image to be a vector in R^3:

All rights reserved by *www.ijsrd.com*

$$p = \begin{bmatrix} p_x \\ p_y \\ p_\theta \end{bmatrix}$$

……..6

Where (p_x ,p_y) is the location of the point, and p_0 is the local orientation of the point (e.g., the direction of the gradient, edge normal, or tangent).

We now need a measure to determine how well these oriented edge maps match. One measure that fulfills these conditions is

$$h_\alpha(M,I) = \max_{m\in M}\min_{i\in I}\max\left\{\left\|\begin{bmatrix} m_x - i_x \\ m_y - i_y \end{bmatrix}\right\|, \frac{|m_\theta - i_\theta|}{\alpha}\right\},$$

…….7

This has the same general form as the previous Hausdorff measure, but the distance between two points is now measured by taking the maximum of the distances in translation and orientation. In this measure α, is a normalization factor that makes the orientation values implicitly comparable with the location values.

Our system discretizes the orientations such that α=1 and uses L_∞ the norm. In this case, the measure for oriented points simplifies to

$$h(M,I) = \max_{m\in M}\min_{i\in I}\|m - i\|_\infty,$$

…..8

H. Probability of false alarm

Let us consider matching a single connected chain of oriented object pixels to the image at some specified location.

For some pixel in the object chain, we will say that it results in a hit if the transformed object pixel matches an image pixel in both location and orientation according to our measure, and otherwise, we will say that it results in a miss. If the object chain is mapped to a sequence of such hits and misses, then this yields a stochastic process. Note that if some pixel in the object chain maps to a hit, this means that locally, the object chain aligns with an image chain very closely in both location and orientation. It is thus very likely that the next pixel will also map to a hit since the chains are expected to continue in the direction specified by the local orientation with little change in this orientation.

Let S_i be a random variable describing whether the *i*th object pixel is a hit or a miss, and let s_i be the value taken by this variable for a specific object chain. If the probability of being in each state at each pixel is dependent only on and the previous state

$$\Pr[S_i = s|(S_{i-1} = s_{i-1}) \wedge \ldots \wedge (S_0 = s_0)] = \Pr[S_i = s|S_{i-1} = s_{i-1}]$$

…9

then the process is said to be a Markov process. If, furthermore, the probability does not depend on , then the process is a Markov chain.

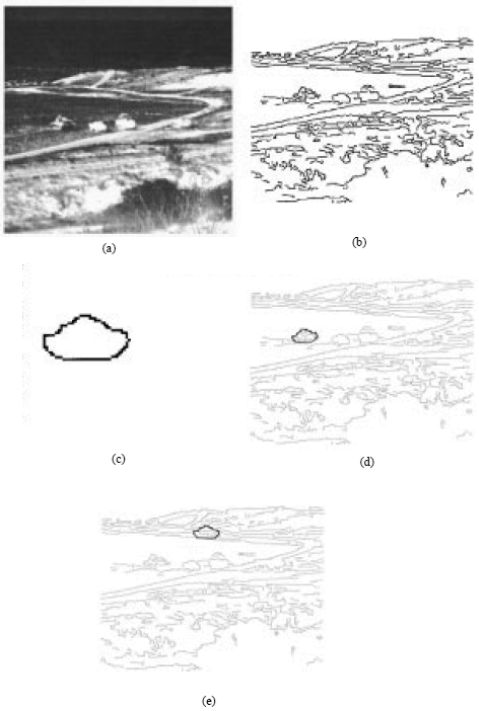

Fig. 6: Target recognition example. (a) FLIR image after histogram equalization. (b) Edges found in the image. (c) Smoothed edges of a tank model. (d) Detected position of the tank. (e) False alarm

II. MATCHING EDGE PIXELS

Here we describe techniques that are used to reduce the running time of the system when there are multiple object models that may appear in the image.

Chamfer matching [3], [4] is an edge matching technique that minimizes the sum of the distances from each object edge pixel to its closest image edge pixel over the space of possible transformations. This technique is closely related to minimizing the generalized Hausdorff measure, which instead minimizes the Kth largest of these distances. Since the chamfer measure sums the distances over all of the object pixels, it is not robust to occlusion. In the original formulation of chamfer matching, Barrow *et al.* [3] used a starting hypothesis and an optimization procedure to determine a position of the model that is a local minimum with respect to the chamfer measure. This method requires a good starting hypothesis to converge to the global minimum. Borgefors [4] proposed a hierarchical method that examines an edge pyramid of the model and image. A number of initial positions are considered at some level of the pyramid, where a Gauss–Seidel optimization procedure is used to find a local minima for each initial position. Poor local minima are rejected. The remaining positions are considered at the next lower level of the pyramid, and the procedure is repeated until local minima are found at the lowest level of the pyramid. This technique performs a search of the image forgood local minima, but it still cannot guarantee that the best transformation is found.

Paglieroni *et al.* [5], [6] have considered methods to speed up the search over all possible transformations in chamfer matching by probing a distance transform of the image at the locations of the transformed object edge pixels. This distance transforms measures the distance of each pixel in the image from an edge pixel and can be computed

All rights reserved by *www.ijsrd.com*

efficiently using a two-pass algorithm [7], [5]. If the sum of the distance transform probes at each of the object pixels at some transformation is large enough, then we can rule out not only this transformation but also many transformations close to it since we know that the close transformations will yield a similar distance transform value for each pixel in the object model. This method is able to search an entire image efficiently and is able to guarantee that the best match (or all matches that surpass some threshold) according to the chamfer measure are found.

Similar techniques have been developed to perform efficient matching using the generalized Hausdorff measure [9], [10], [8], which is robust to partial occlusions of the object. First, the image is dilated by E_δ(as described in the previous section), and the distance transform of this dilated image is determined. If the Kth largest probe into this distance transform is 0, then a match of size (at least) K has been found. Otherwise, the Kth largest probe yields the distance to the closest possible position of the object model that could produce a match of size . We can thus rule out any transformation that does not move any object pixel more than this distance. To improve efficiency, the transformation space is discretized, but to ensure that no good matches are missed, this discretization is such that adjacent transformations do not map any object pixel more than one pixel (Euclidean distance) apart in the image. Now, if d is the value of the Kth largest probe, we can ruleout at least those transformations with a city-block distance (L_1 norm) less than d from the current transformation in the discretized transformation space since such transformations are guaranteed to move each object pixel less than d pixels from the current location.

A. Probability of false detection

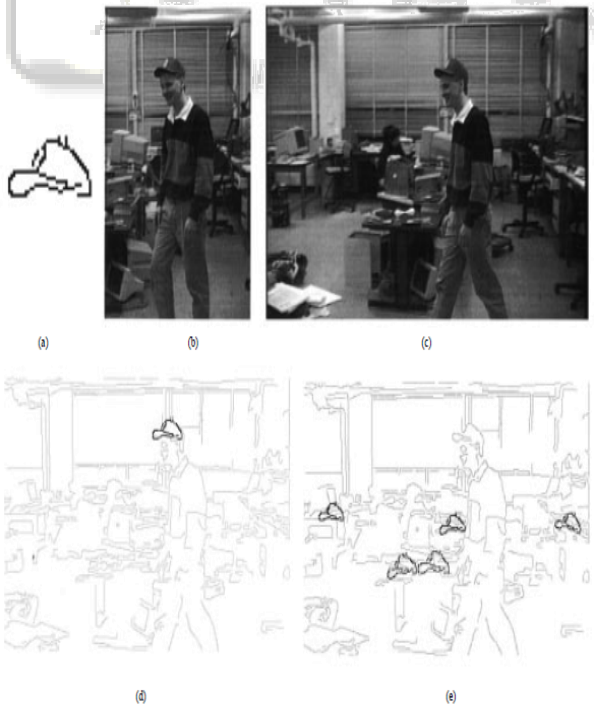

Fig 7: Image sequence example. (a) Object model. (b) Part of the image frame. (c) Image frame in which we are searching for the model. (d) Position of the model located. No false alarms were found. (e) Several false alarms were found

In matching edge pixel, we consider an example of the use of these techniques in a complex indoor scene as shown in the fig 7. In this example an object model is selected, now considering a part of the image frame from which the model is extracted. Fig 7(c) in which we are searching for a model in image frame. The position of the model is located using matching edge pixel. Now considering two cases, one in which no false alarms were found as shown in fig 7(d) but sometimes several false alarms were found when orientation information was not used. This each yielded a higher score than the correct position of the model.

When orientation information was not used, several positions of the object were found that yielded a better score than the correct position of the object hence in comparison to the matching oriented edge pixel there are less false alarms.

B. Block-matching motion detection

To detect motion by identifying pairs of pixel neighborhoods, in the two successive images, that has a similar grey level distribution. Pixels in the first image can, as before, be preselected to reduce the amount of data to process. This can be done using background removal (pixels in the background are not expected to move) or by computing the difference between the two images (a direct indication of motion).

The objective is then to compute a velocity field for each preselected pixel (x,y) in the first image by defining a small neighbourhood (or *pixel block*, typically 10x10) centred at position (x,y). A *search* block, also centred at (x,y), is defined in the second image. The size of the search block is determined by the maximum displacement expected in the given frame interval. Then, all possible pixel blocks in the search area are compared with that in the first image using a similarity function (e.g. sum of the pixel-to-pixel absolute difference). Under ideal conditions of no changes in illumination and object shape, a *matching* block would be found where the similarity function is zero. Under more realistic conditions, a match is defined as the block that produces the minimum similarity value. A threshold is applied to account for cases of drastic changes in shape or illumination or that of objects leaving the scene.

As an example, Fig. 8 shows a pixel block in a first image and a search block in the second image (sizes have been exaggerated for illustration purposes). The two images are separated by an interval of 0.12 seconds.

Fig. 8: Motion calculation by "block matching"

Irregular motion, movements of arms, legs, and clothing and localized variations in brightness all cause errors in the computed motion vectors compared to the actual overall motion of the individuals in the crowd.

All rights reserved by *www.ijsrd.com*

C. Probability of false alarm:

To compensate for such effects (which can be effectively regarded as zero-mean noise added to the motion vector estimates), the computed vectors are aggregated over small disjoint neighborhoods of 10x10 pixels throughout the image. A typical resulting vector field superimposed on the first image is shown in Fig 9.

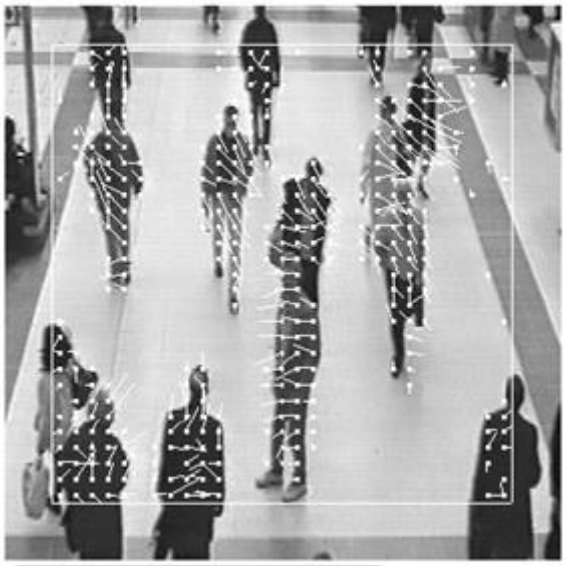

Fig 9: A typical result after calculating motion by "block matching".

Reference to the original images confirms a correlation between these measurements and manual observations. Pixel pre-selection by background removal, and to a lesser extent interframe difference, results in a reduced number of mismatches compared to pre-selection based on thinned edges [11,12].

The block matching technique described here involves substantial computation. However, the commercial demands for video-data compression are such that specially-designed VLSI components are already being developed for fast implementation of these algorithms. It is therefore feasible to use these techniques for real-time crowd velocity estimation.

III. CONCLUSION

Fig. 6 shows an example of the use of matching oriented edge pixel techniques. The image is a low contrast infrared image of an outdoor terrain scene. Fig. 6 also shows a false alarm that was found. Note that the image window for this false alarm is denser with edge pixels than the correct location. Also in matching edge pixel technique which is limited to target recognition. Fig. 7 shows an example of the use of this technique in a complex indoor scene. In this case, the object model was extracted from a frame in an image sequence, and it is matched to a later frame in the sequence (as in tracking applications). Since little time has passed between these frames, it is assumed that the model has not undergone much rotation out of the image plane, and thus, a four-dimensional (4-D) transformation space is used, consisting of translation, rotation in the plane, and scale. The position of the object was correctly located when orientation information was used. Still some false alarms were also found for this case.

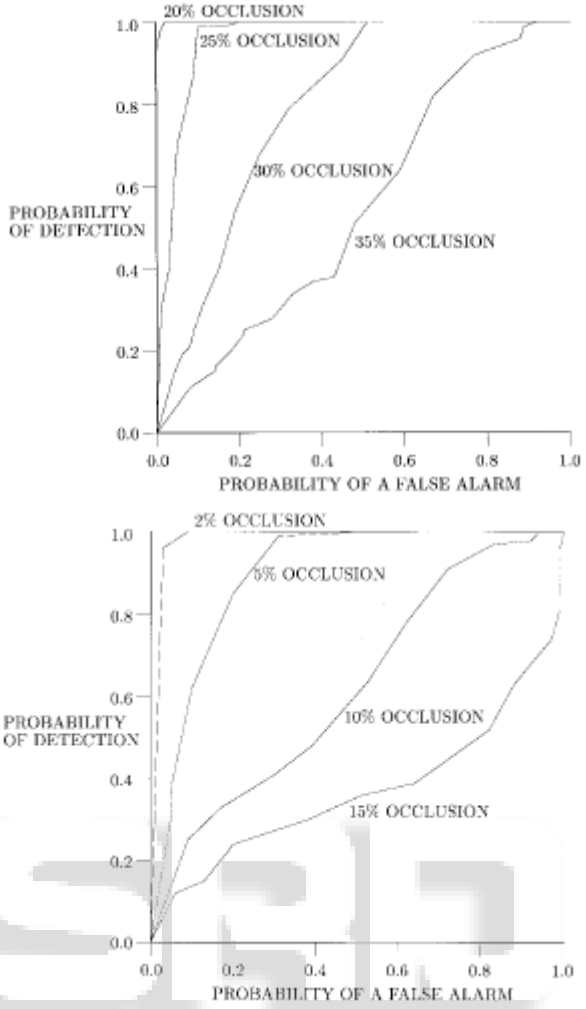

Fig 10: Receiver operating characteristic (ROC) curves generated using synthetic data. (a) ROC curves when using matching edge pixel. (b) ROC curves when using block matching motion detection

We have generated ROC curves for this system using synthetic edge images. Each synthetic edge image was generated with 10% of the pixels filled with random image clutter (curved chains of connected pixels). An instance of a target was placed in each image with varying levels of occlusion generated by removing a connected segment of the target boundary. Random Gaussian noise was added to the locations of the pixels corresponding to the target. An example of such a synthetic image can be found and orientation information was used and when it was not in Fig. 10(a). Fig. 11(b) shows ROC curves generated for cases when block matching technique is used. These ROC curves show the probability that the target was located versus the probability that a false alarm of this target model was reported for varying levels of the matching threshold. When orientation information was used, the performance of the system was very good in these images up to 25% occlusion of the target. On the other hand, when orientation information was not used, the performance degraded significantly before 10% occlusion of the object was reached. Hence according to the proposed technique in this paper, it is possible to use well-established image processing techniques for monitoring and collecting data on crowd behavior.

All rights reserved by *www.ijsrd.com*

REFERENCE

[1]. Hentschel T., 1993, "Image Processing Techniques for the Estimation of Features of Crowd Behaviour in Urban Environments", MSc. Dissertation, King's College London, UK.
[2]. B. Bhanu, "Automatic target recognition: State of the art survey," IEEE Trans. Aerosp. Electron. Syst., vol. AES-22, pp. 364–379, July 1986.
[3] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf, "Parametric correspondence and chamfer matching: Two new techniques for image matching," in Proc. Int. Joint Conf. Artificial Intell., 1977, pp. 659–663.
[4] , "Hierarchical chamfer matching: A parametric edge matching algorithm," IEEE Trans. Pattern Anal. Machine Intell., vol. 10, pp. 849–865, Nov. 1988.
[5] D. W. Paglieroni, "Distance transforms: Properties and machine vision applications," CVGIP: Graphical Models Image Processing, vol. 54, no. 1, pp. 56–74, Jan. 1992.
[6] D. W. Paglieroni, G. E. Ford, and E. M. Tsujimoto, "The positionorientation masking approach to parametric search for template matching," IEEE Trans. Pattern Anal. Machine Intell., vol. 16, pp. 740–747, July 1994.
[7]. A. Rosenfeld and J. Pfaltz, "Sequential operations in digital picture processing," J. Assoc. Comput. Mach., vol. 13, pp. 471–494, 1966.
[8] W. J. Rucklidge, "Locating objects using the Hausdorff distance," in Proc. Int. Conf. Comput. Vision, 1995, pp. 457–464.
[9] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge, "Comparing images using the Hausdorff distance," IEEE Trans. Pattern Anal. Machine Intell., vol. 15, pp. 850–863, Sept. 1993.
[10] D. P. Huttenlocher and W. J. Rucklidge, "A multi-resolution technique for comparing images using the Hausdorff distance," in Proc. IEEE Conf. Comput. Vision Patt. Recogn., 1993, pp. 705–706.
[11] Velastin S.A., Yin J.H, Davies A.C., Vicencio-Silva M.A., Allsop R.E. and Penn A., 1994: "Automated Measurement of Crowd Density and Motion using Image Processing", 7th IEE International Conference on Road Traffic Monitoring and Control, 26-28 April 1994, London, UK, 127-132
[12].Velastin S.A., Yin J.H., Vicencio-Silva M.A., Davies A.C., Allsop R.E. and Penn A., 1994: "Image Processing Techniques for On-line Analysis of Crowds in Public Transport Areas", IFAC Symposium on Transportation Systems: Theory and Application of Advanced Technology, 24-26 August 1994, Tianjin, China

All rights reserved by *www.ijsrd.com*