THE INTERNATIONAL JOURNAL OF
SCIENCE & TECHNOLEDGE

Moving Object Detection Based on Fuzzy Color Histogram Features and Dynamic Threshold Optimization

S. Kannan
Department of Electronics and Communication Engineering
Regional Office of Anna University, Madurai, Tamil Nadu, India
A. Sivasankar
Department of Electronics and Communication Engineering
Regional Office of Anna University, Madurai, Tamil Nadu, India

Abstract:
This paper proposes an efficient motion detection and crowd counting system based on background subtraction using dynamic threshold and fuzzy logic. Here two methods are used effectively for object detection followed by people counting and compare these performance based on accurate estimation. In dynamic threshold based object counting, morphological process and filtering are used effectively for unwanted pixel removal from the background. Along with this dynamic threshold, we introduce a background subtraction algorithm for temporally dynamic texture scenes using a clustering-based feature, called fuzzy color histogram (FCH), which has an ability of greatly attenuating color variations generated by background motions while still highlighting moving objects for efficient people counting. Experimental results demonstrate that proposed method is effective for motion detection and crowd counting system based on background subtraction using dynamic threshold and fuzzy logic, compared to several other competitive methods.

1. Introduction

Detection, tracking and counting of humans in video sequences is important for a number of applications, such as video surveillance and human-computer interaction. Study of moving objects is an interesting subject of research in computer vision. There are few common moving object detection methods such as, Inter-frame differencing, background subtraction, optical flow, feature point statistics and classification, and feature matching and tracking [1][2][3]. Background subtraction method is an important first step for many vision problems in which the object is separated from background clutter, by comparing the motion patterns, and assists subsequent higher-level operations, such as tracking, object recognition, etc. Background subtraction algorithms are expected to be robust both in the short term and throughout the lifetime of the vision system, because the environment can change substantially [4][5].

Dynamic Threshold Optimization (DTO) is used to detect objects in moving environment. Though DTO is an algorithm independent technique it will be more effective to be used with any search and optimization algorithm.

We propose a simple and robust method for background subtraction in dynamic texture scenes. The underlying principle behind our model is that color variations generated by background motions are greatly attenuated in a fuzzy manner. Therefore, compared to preceding methods using local kernels, the future method does not require estimation of any parameters (i.e., nonparametric). This is quite advantageous for achieving the robust background subtraction in a wide range of scenes with spatiotemporal dynamics. Specifically, we propose to get the local features from the fuzzy colour histogram (FCH). Then, the background model is dependably constructed by computing the similarity between local FCH features with an online update procedure [6] [7]. To verify the advantage of the proposed method, we finally compare ours with competitive background subtraction models proposed in the literature using various dynamic texture scenes.

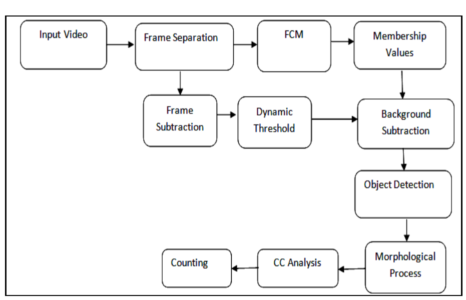

Figure 1: Block diagram

2. Fuzzy Color histogram features

In this paper, the colour histogram is viewed as a colour distribution from the probability viewpoint. Given a colour space containing colour bins, the colour histogram of image containing pixels is represented as, where is the probability of a pixel in the image belonging to the i^{th} colour bin, and is the total number of pixels in the colour bin. According to the total probability theory, can be defined as follows:

$$h_i = \sum_{j=1}^{N} P_{i|j} P_j = \frac{1}{N} \sum_{j=1}^{N} P_{i|j} \qquad \text{----- (1)}$$

Where P_j is the probability of a pixel selected from image I being the j^{th} pixel, which is 1/N, and $P_{i/j}$ is the conditional probability of the selected j^{th} pixel belonging to the i^{th} colour bin. In the context of CCH, is defined as

In the context of CCH, is defined as

$$P_{i|j} = \begin{cases} 1, & \text{if the } j\text{th pixel is quantized into the } i\text{th color bin} \\ 0, & \text{otherwise.} \end{cases} \qquad \text{----- (2)}$$

This definition leads to the boundary issue of CCH such that the histogram may undergo abrupt changes even though colour variations are actually small. This reveals the reason why the CCH is sensitive to noisy interference such as illumination changes and quantization errors. The proposed FCH essentially modifies probability $P_{i|j}$ as follows. Instead of using the probability $P_{i|j}$, we consider each of the N pixels in image I being related to all the colour bins via fuzzy-set membership function such that the degree of "belongingness" or "association" of the j^{th} pixel to the i^{th} colour bin is determined by distributing the membership value of the j^{th} pixel, μ_{ij} to the i^{th} colour bin.

2.1. Definition (Fuzzy Colour Histogram)

The fuzzy colour histogram (FCH) of image I can be expressed as F(I)=[f1,f2, f3,...... fn], where

$$f_i = \sum_{j=1}^{N} \mu_{ij} P_j = \frac{1}{N} \sum_{j=1}^{N} \mu_{ij}. \qquad \text{----- (3)}$$

has been defined in (1), and is the membership value of the j^{th} pixel in the i^{th} colour bin. In contrast with CCH, our FCH considers not only the similarity of different colours from different bins but also the dissimilarity of those colours assigned to the same bin. Therefore, FCH effectively alleviates the sensitivity to the noisy interference [8][9].

2.2. FCM Clustering Technique

The Fuzzy C Means (FCM) clustering is the combination of fuzzy algorithm, C Means clustering, and thresholding algorithm. The goal of clustering analysis is to divide a given set of data or objects into a cluster, which represents subsets or a group. The partition must have two properties, homogeneity inside clusters and heterogeneity between the clusters [8].

2.3. Clustering

In our work, a partition, P is a set of disjoint subsets of E and the element Ps of P is called 'cluster' and the centers of the clusters are called 'centroids' or prototypes. The membership functions do not reflect the actual data distribution in the input and output spaces. To build membership function from the data available, a clustering technique may be used to partition the data, and then produce membership functions from the resulting clustering [8][9].

2.4. C Means

Many techniques have been developed for data clustering. In this paper c-means clustering is used for data grouping or classification when the number of the clusters is known. The technique consists of the following steps:

Step 1: Choose the number of clusters-K

Step 2: Set initial centers of cluster C_1, C_2……C_k.

Step 3: Classify each vector X1= [x_{11}, x_{12}…… x_{1n}] T into the Closest center C_i by Euclidean distance measure: $\| xi - ci \| = \min \| x_i - c_i \|$

Step 4: Recomputed the estimates for the cluster enter C_i; let $C_i = [C_{i1}, C_{i2}, \ldots C_{in}]T$

C_{in} be computed by: Where Ni is the Number of vectors in the i-th cluster

Step 5: If more of the cluster center (C_i = 1,2,…,k) in step 4 stop; otherwise go to step 3

The condition function used for fuzzy C- means clustering is;

$$J(v) = \sum_{i=1}^{c} \sum_{k=1}^{n} U_{ik}^{n} \left| x_k - v_i \right|^2 \qquad \text{------ (4)}$$

Where X_1 ... X_n - „ n - data sample vector, V_1…V_c - „ c - denotes cluster centers (centroids); U= U_{ik} cxm matrix, where U_{ik} is the i-th membership value of the k-th input sample x_k, and The membership values satisfy the following conditions:

$0 < u_{ik} < 1$ where., i = 1,.....,c and k = 1,…,n

$$\sum_{i=1}^{c} u_{ik} = 1; \quad k = 1,\ldots n; \qquad \text{------ (5)}$$

$$0 < \sum_{k=1}^{n} u_{ik} < 1; \quad i = 1,..c; \qquad \text{------ (6)}$$

$m \in (1, \infty)$ is an exponent weight factor

Where,

$$P(\omega_i \mid \mathbf{x}_j) = u_{ij} = \frac{\left(\frac{1}{d_{ij}}\right)^{1/(b-1)}}{\sum_{r=1}^{c}\left(\frac{1}{d_{rj}}\right)^{1/(b-1)}} \qquad \text{----- (7)}$$

Where $d_{ij} = || x_i - c_i||^2$. These membership values are needed to be computed once and stored as a membership matrix in advance. Therefore, we can easily build FCH for the incoming video frame by directly referring to the stored matrix without computing membership values for all pixels. We finally define the local FCH feature vector at the each pixel position of the each video frame for the robust background subtraction in dynamic texture scenes, as follows:

$$\mathbf{F}_j(k) = (f_{j,1}^k, f_{j,2}^k, \ldots, f_{j,c}^k), \quad f_{j,i}^k = \sum_{q \in W_j^k} u_{iq} \qquad \text{----- (8)}$$

The Membership value indicates the color feature computed at the pixel position to the color bin as cited. By using the difference of our local features between consecutive frames, we can build the reliable background model. Local CCHs obtained from the same pixel position of two video frames are quite different due to strongly waving leaves. In contrast to that, we confirm that FCH provides relatively consistent results even though dynamic textures are widely distributed in the background. Therefore, it is thought that our local FCH features are very useful for modeling the background in dynamic texture scenes. The background subtraction will be performed based on the similarity measure of local FCH features for foreground and background detection [8].

3. Background Subtraction

This method gives the procedure of background subtraction based on our local FCH features. An observed FCH vectors are compared with the model FCH vector for classifying a given pixel into either background or moving objects in the current frame.

$$B_j(k) = \begin{cases} 1, & \text{if } S(\mathbf{F}_j(k), \widehat{\mathbf{F}}_j(k)) > \tau, \\ 0, & \text{otherwise,} \end{cases} \qquad \text{----- (9)}$$

where, $B_j(k) = 1$.,denotes that the pixel in the current video frame is determined as the background whereas the corresponding pixel belongs to moving objects if $B_j(k) = 0$. T is a thresholding value ranging from 0 to 1. The similarity measure adopts normalized histogram intersection for simple computation, is defined as follows:

$$S(\mathbf{F}_j(k), \widehat{\mathbf{F}}_j(k)) = \frac{\sum_{i=1}^{c} \min\left[f_{j,i}^k, \widehat{f}_{j,i}^k\right]}{\max\left[\sum_{i=1}^{c} f_{j,i}^k, \sum_{i=1}^{c} \widehat{f}_{j,i}^k\right]} \qquad \text{----- (10)}$$

Where, $F_j(k)$ denotes the background model of the each pixel position in the current video frame.

4. Object Detection Based on Dynamic Threshold Method

After the background image *B(x, y)* is obtained, subtract the background image *B(x,y)* from the current frame $F_k(x, y)$. If the pixel difference is greater than the set threshold *T*, then it is determined as the pixels appear in the moving object, or else, as the background pixels. The moving object in the video sequence can be detected once the threshold operation is done. Its expression is as follows:

$$D_k(x, y) = \begin{cases} 1 & |F_k(x, y) - B_{k-1}(x, y)| > T \\ 0 & others \end{cases}$$

where $D_k(x, y)$ is the binary image of differential results and *T* is gray-scale threshold; its size determines the accuracy of object detection. Like in the algorithm *T* is a fixed value, and is suitable only for an ideal situation, and is not suitable for complex environment with lighting changes. Hence, this paper proposes the dynamic threshold method, which dynamically changes the threshold value according to the lighting changes of the two images obtained. So a dynamic threshold *T* is added to the above algorithm and the mathematical expression is obtained as follows:

$$\Delta T = \lambda \bullet \frac{1}{M \times N} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} |F(i, j) - B(i, j)| \qquad \text{----- (11)}$$

Then,

$$D_k(x,y)=\begin{cases}1 & |F_k(x,y)-B_{k-1}(x,y)|>T+\Delta T\\ 0 & others\end{cases}$$ ----- (12)

where A is the inhibitory coefficient. It is set to a value according to the requirements of practical applications. Usually the reference value is 2. M x N is the size of each image to contend with. M x N numerical results indicate the number of pixels in detection region. T' reflects the overall changes in the environment. If small changes in image illumination, dynamic threshold T' takes a very small value. Under the premise of enough pixels in the detection region, T' will tend to 0. If the image illumination changes significantly, then the dynamic threshold T' will also increase similarly significant. Thus this method can effectively suppress the impact of light changes.

5. Morphological Filtering Process

Mathematical morphology (MM) is a theory and technique for the analysis and processing of geometrical structures. It is based on set theory, lattice theory, topology, and random functions. Though MM is most commonly applied to digital images it can also be employed as well on graphs, surface meshes, solids, and many more other spatial structures.

Topological and geometrical continuous-space concepts like size, shape, convexity, connectivity, and geodesic distance, can be analyzed by MM on both continuous and discrete spaces. It is also noted that MM is the foundation of morphological image processing, which consists of a set of operators that transform images according to the above characterizations.

MM was originally developed for binary images, and later on was extended to grayscale functions and images. The successive generalization to complete lattices is widely accepted today as MM's theoretical foundation.

6. Connected Component Analysis

The output of the change detection module is the binary image that contains only two labels, i.e., '0' and '255', representing as 'background' and 'foreground' pixels respectively, with some noise. The goal of the connected component analysis is to detect the large sized connected foreground region or object which is one of the important operations in motion detection. The pixels that are jointly connected can be clustered into changing or moving objects by analyzing their connectivity.

In binary image analysis, the object is extracted using the connected component labeling operation, which consist of assigning a distinctive label to each maximally connected Foreground region of pixels. One of the important labeling approaches is "classical sequential labeling algorithm". It is based on two faster scan of binary image. The first scan performs the temporary labeling to each foreground region pixels by checking their connectivity of the scanned image. When a foreground pixel with two or more than two foreground neighboring pixels carrying the same label is found, the labels associated with those pixels are registered as being equivalent. That means these regions are from the same object. The handling of equivalent labels and merging thereafter is the most complex task.

The first scan gives temporary labels to the foreground pixels according to their connectivity. The connectivity check can be done with the help of either a 4-connectivity or 8-connectivity approach. 8-connectivity approach is used. Here, the idea is to label the whole blob at a time to avoid the label redundancies.

The labeling operation scans the image moving along the row until it comes to the point P, for which S = {255}. When this is true, it checks the four neighbours of which Based on that information, the labeling of *P* occurs as follows,

If all four neighbours are '0' assign a new label to P, and increment the label,
Else
If only one neighbour has S = {255} assigns its label to *P*
Else (i.e., more than one of the neighbours has S = {255})
Assign one of the labels to P.

Here, note that the relation between the pixels that are expressed through a "label value" in the labeled image depends on the value of the label. Background, labeled as l_B is not necessarily to be connected, but the two pixels labeled l_P from the foreground region are to be connected.

7. Results and Discussion

The given input video is converted into its respective frames. The below image consists of frame sequence of a video.

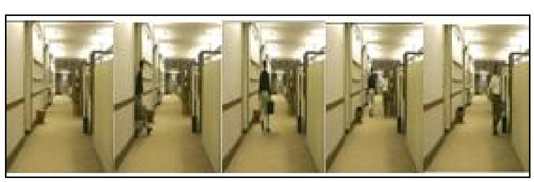

Figure 2: Input Video Frames

It represents the object class which is human beings. In this the given input video is converted into frames for object detection and counting purpose based on fuzzy color histogram.

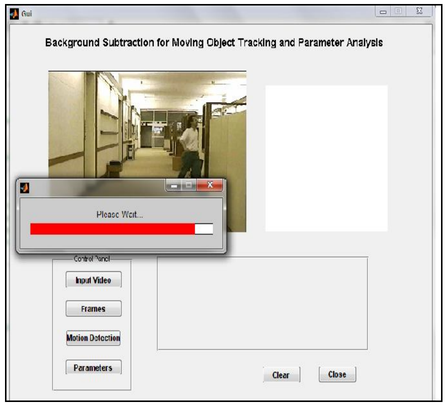

Figure 3: Frame conversion

An Input Video (.avi files) is converted into still images for processing it and detects the moving objects. These sequences of images gathered from video files by finding the information about it through 'aviinfo' command. These frames are converted into images with help of the command 'frame2im' Create the name to each images and this process will be continued for all the video frames.

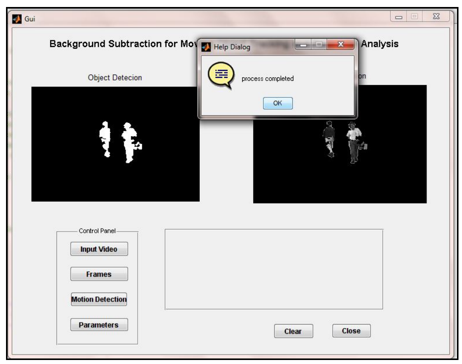

Figure 4: Motion Detection

After the background image *B(x, y)* is obtained, subtract the background image *B(x,y)* from the current frame F_k *(x, y)*. If the pixel difference is greater than the set threshold *T*, then determines that the pixels appear in the moving object, otherwise, as the background pixels.

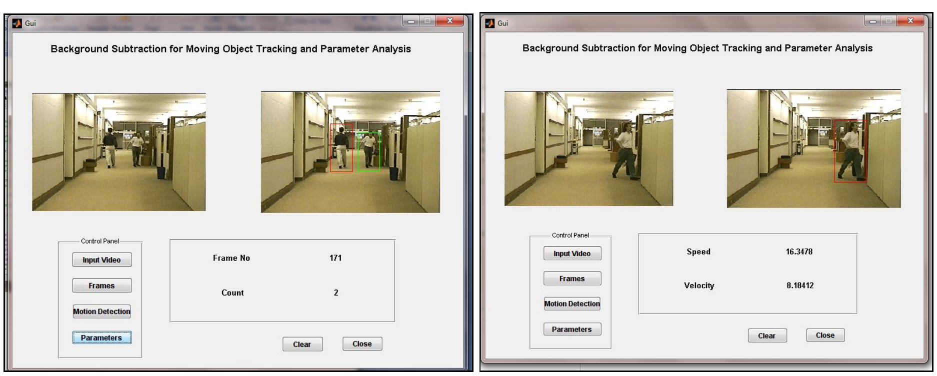

Figure 5: Parameter analysis

8. Conclusion

In this project we have proposed a background subtraction for object detection and counting based on fuzzy color histogram. Background subtraction method is very accurate for moving object detection and counting from dynamic texture as comparing with background subtraction with through simple threshold method. The proposed system effectively detects human motion and counts the number of persons present in the video sequence, precisely. This system also helps finding the velocity and speed of the process.

Though this method of Human detection is effective, there are few drawbacks present in this system such as interference of shadow. In future system, contour projection analysis can be combined with the shape analysis to remove the effect of shadow, and thus the moving human bodies are accurately and reliably detected.

9. References

1. L. Dong, V. Parameswaran, V. Ramesh, and I. Zoghlami, "Fast crowd segmentation using shape indexing," in Proc. IEEE Int. Conf. Comput. Vis., 2007, pp. 1–8.
2. B. Leibe, E. Seemann, and B. Schiele, "Pedestrian detection in crowded scenes," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2005, vol. 1, pp. 875–885.
3. T. Zhao, R. Nevatia, and B. Wu, "Segmentation and tracking of multiple humans in crowded environments," IEEE Trans. Pattern Anal. Mach. Intell., vol. 30, no. 7, pp. 1198–1211, Jul. 2008.
4. B. Chan, V. Mahadevan, and N. Vasconcelos, "Generalized Stauffer-Grimson background subtraction for dynamic scenes," Mach. Vis. Applicat., vol. 22, no. 5, pp. 751–766, 2011.
5. G. Dalley, J. Migdal, and W. Grimson, "Background subtraction for temporally irregular dynamic textures," in Proc. IEEE Workshop on Applications of Computer Vision (WACV), Jan. 2008, pp. 1–7.
6. D. Navneet and T. Bill , "Histograms of Oriented Gradients for Human Detection," International Conference on Computer Vision & Pattern Recognition (CVPR '05) 1, 886—893, 2005.
7. M. Piccardi, "Background subtraction techniques: a review," IEEE International Conference on Systems, Man and Cybernetics, 3099-3104, 2004.
8. J. Han and K. Ma , "Fuzzy Color Histogram and Its Use in Color Image Retrieval," IEEE Transactions on Image Processing, Vol. 11, No. 8, August 2002.
9. M.Sivabalakrishnan and D.Manjula, "Adaptive Background subtraction in Dynamic Environments Using Fuzzy Logic," International Journal of Video& Image Processing and Network Security IJVIPNS-IJENS Vol.10 No.01, 13-16