Video Object Extraction Based on a Comparative Study of Efficient Edge Detection Techniques

Kavitha Ganesan[1] and Shanmugam Jalla[2]
[1]Lecturer Electronics Department, Anna University, India
[2]Principal Tagore Engineering College, India

Abstract: *Segmentation of objects serves as a key of image analysis and pattern recognition. The main focus of the paper is related to the development of efficient algorithms for automatic segmentation of objects in image sequences. Fast, automatic and robust segmentation are necessary in many aspects of multimedia applications due to its capability that it can automatically detect objects appearance and in addition it can be used for object tracking system. The extraction of semantically meaningful video object for tracking and surveillance application can be obtained in the process of implementing the proposed algorithms. Further in this paper a comparative study between video object extraction based on change detection and model matching techniques is given. Experimental results prove the effectiveness of the proposed algorithms.*

Keywords: *Segmentation, video object extraction, change detection, model matching.*

Received August 23, 2007; accepted November 7, 2007

1. Introduction

Video segmentation has received more attention with the increasing popularity of content-based video coding schemes and interactive multimedia applications. The new standard MPEG-4 enables content-based interactivity by introducing the concept of Video Object Planes (VOP).The frames of the input sequence are decomposed into video object planes such that each VOP describes a particular object of interest. A number of techniques and algorithms have been proposed for detecting moving objects in video in the past, each of which has its particular features and applications. Towards meeting this challenge, several approaches were introduced for segmentation of video sequences into moving video objects. They can be broadly classified into three categories: spatio-temporal, motion, morphological and model-matching techniques.

Spatio-temporal segmentation techniques [11] attempt to identify the objects in a scene based on spatial and temporal information without explicitly computing the motion parameters. Motion segmentation techniques [1] rely on motion parameters, explicitly computed from the spatial color/luminance information. Morphological techniques [10] for video object segmentation, which involve morphological filters or watershed segmentation techniques are computationally efficient and have gained increased popularity in recent years. Starting with an initial object model, model-matching segmentation techniques [7] aim to locate the object in the video scene based on the best match between a model of the object and the scene's frames.

This paper is organized as follows. Section 2 provides an overview of the approach based on change detection technique and the method by which moving objects can be extracted from the moving pixels obtained by spatial temporal entropy. The resulting video objects have also been discussed. In section 3, the model matching based method for more accurate extraction of the video object planes has been analyzed in detail and experimental results demonstrating the application of the proposed algorithm to video sequences are shown. Finally, section 4 provides conclusions based on the proposed approaches and the experimental results have been compared.

2. Change Detection Based Video Object Extraction

A desirable video object extraction scheme for content-based applications should meet the following criteria:

- Segmented object should conform to human perception i.e., semantically meaningful objects should be segmented.
- Segmentation algorithm should be efficient and achieve fast speed.
- Initialization should be simple and easy for users to operate (human intervention should be minimized). One feasible solution that satisfies these criteria is edge change detection.

2.1. Edge Change Detection

In Video Object (VO) segmentation methods, which are using mathematical morphology and perspective motion model, objects of interest should be initially outlined by human observer. From the manually specified object boundary, the correct object boundary is calculated using a morphological segmentation tool. The obtained VOP is then automatically tracked and updated in successive frames. It has difficulty in dealing with a large non rigid object movement and in the presence of occlusion [2], especially in the VOP tracking schemes.

The algorithm based on edge change detection allows automatic detection of the new appearance of a VO. The edge change detection for inter-frame difference is another stream of popular schemes because it is straightforward to implement and enables automatic detection of new appearance. This ability enables to develop a fully automated object-based system, such as an object-based video surveillance system. It is found that the algorithms based on inter-frame change detection render automatic detection of objects and allow larger non rigid motion compared to mathematical morphology and perspective motion model methods. The drawbacks are small false regions detected by decision error due to noise. Thus, small whole removal using morphological operation and removal of false parts like uncovered background by motion information are usually incorporated. Another drawback in edge change detection is that object boundaries are irregular in some critical image areas, which must be smoothened and adapted by spatial edge information. Since spatial edge information is useful for generating VOP with accurate boundaries, a simple binary edge difference scheme may be assumed to be a good solution. In order to overcome boundary inaccuracy multiple features, multiple frames and spatial-temporal entropy methods are used. In addition, it gives robustness to noise and occluding pixels.

In this automatic VO segmentation algorithm [4, 5] edge change detection starts with edge detection which is the first and most important stage of human visual process. Edge information plays a key role in extracting the physical change of the corresponding surface in a real scene, exploiting simple difference of edges for extracting shape information of moving objects in video sequence suffers from great deal of noise even in stationary background. This is due to the fact that the random noise created in one frame is different from the one created in the successive frame, and thus results in slight changes of the edge locations in the successive frames. Thus difference edge of frames suppresses the noise in luminance difference by means of canny edge detector [3]. The difference of edges is calculated by equation 1.

$$\left|\Phi\left(I_{n-1}\right)-\Phi\left(I_{n}\right)\right|=\left|\theta\left(\nabla G * I_{n-1}\right)-\theta\left(\nabla G * I_{n}\right)\right| \quad (1)$$

where, the edge maps Φ(I) are obtained by the canny edge detector, which is accomplished by performing a gradient operation on the Gaussian convoluted image, followed by applying the non maximum suppression to the gradient magnitude to thin the edge and the threshold operation with hysteresis to detect and link edges.

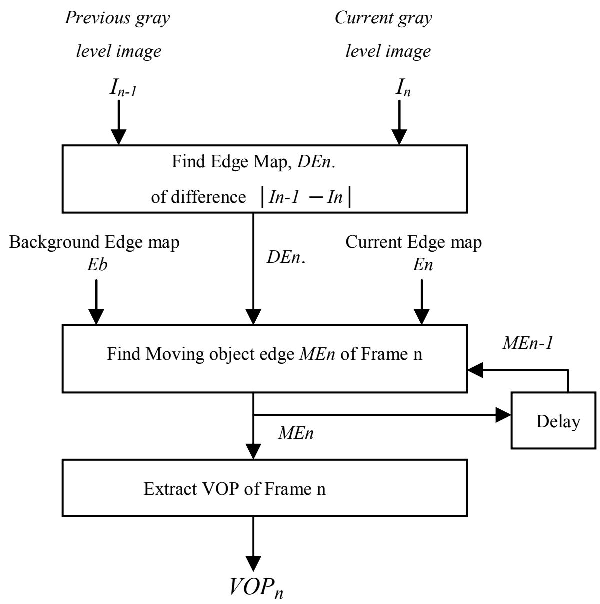

Figure 1. Block diagram of VOP extraction based on Change detection

The moving edge of the current frame are extracted based on the current frame edge map $En = \Phi\ (I_n)$, where Φ represent canny edge detector, background edge map E_b , and difference edge DE_n this is shown in Figure 1. Where E_b contains absolute background edges in case of a still camera and can be extracted from the first frame or by counting the number of background edge occurrence for each pixel through the first several frames. Edge model $E_n= \{e_1,....,e_k\}$ is defined as a set of all edge points detected by the canny operator in the current frame n. Similarly, $ME_n = \{m_1,....,m_l\}$, is defined as the set of l moving edge points, where ME_n represent the moving edges of frame n, where $l \leq k$ and $ME_n \subseteq E_n$. The edge points in ME_n are not restricted to the moving object's boundary, and can be in the interior of the object boundary. If DE_n denotes the set of all pixels belonging to the edge map from the difference image, then the moving edge map generated by edge change is given by selecting all edge pixels within a small distance T_{change} of DE_n as shown in equation 2.

$$ME_n^{change} = \{e \in E_n \mid \min \|e-x\| \leq T_{change}\} \quad (2)$$

Some ME_n might have scattered noise which needs to be removed before proceeding to the next steps. In addition, a previous frame's moving edges can be referenced to detect temporarily still moving edges. Temporarily still moving edges of the present frame is calculated by equation 3. This equation states that still moving edge points belongs to the edges of current frame E_n and it should not belong to the background edges E_b. Still moving edges are obtained by selecting all edge pixel within a small distance T_{still} of ME_{n-1}, where ME_{n-1} is the previous frame moving edge

$$ME_n^{still} = \{e \notin E_n \mid \min \|_{x \in ME_{n-1}} e - x\| \leq T_{still}\} \quad (3)$$

The final moving edge map for current frame is expressed by combining the two maps

$$ME_n = ME_n^{change} \cup ME_n^{still} \quad (4)$$

As change detection suffers from boundary in accuracy we additionally use spatial-temporal entropy [6] to detect the moving pixels. In this method entropy of each pixel in a frame is calculated. Based on the entropy value moving pixels are determined.

2.2. Spatial Temporal Entropy

Background subtraction is a more popular method for motion detection. In this background model estimation is done to determine the absolute background. A simple background model can be the average image over some training period. However when the scene is more complex, for example there are moving leaves or waves, estimated background model cannot represent the background very well. And more over background subtraction suffer from boundary inaccuracy. To solve this problem we go for spatial temporal entropy [8]. Although entropy is the concept of information theory, it is widely used in image segmentation. Entropy is a measure of uncertainty. In image sequences, the pixel value at a particular location could change from frame to frame due to the following reasons:

- The camera channel noise and noises brought by flickering of light
- There are moving objects which make the pixel value change from background to object or from object to background.

In this change of pixel value from frame to frame is considered as the state transition of pixel, e.g., in a 256 level gray image, each pixel has 256 states. Along with time, the pixel's state would change from one to another. Pixel state's change brought by noises would be in a small range, but those brought by motion will be large. So the diversity of state at each pixel can be used to characterize the intensity of motion at its position. In order to get the diversity of pixel state accumulation window is used, where spatial state variation and temporal variation of a pixel are accumulated. An accumulating window is a rectangular spatial window, used to reflect the relationship between a pixel and its neighborhood. Figure 2 shows accumulating window for L frames. In this pixel (i, j) is used to form the spatial-temporal histogram. Here $W * W *$ L pixels are accumulated to form the histogram of pixel (i, j). Once the histogram is obtained, the corresponding probability for each pixel can be computed by equation 5.

$$P_{i,j,q} = \frac{H_{i,j,q}}{N} \quad (5)$$

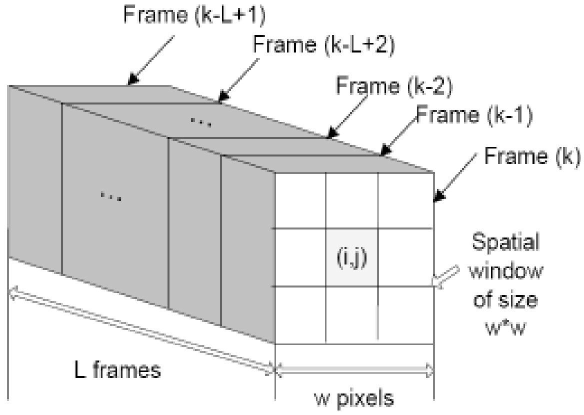

Figure 2. Accumulation window.

In equation 2, N is the total number of pixels in the histogram, $H_{i,j,q}$ denotes number of pixels in histogram bin, q represent the bin of histogram. Total number of bin in the histogram is determined by the states variation of pixel, where $\sum P_{i,j,q} = 1, q = 1, \ldots, Q$, where Q is the total number of bins in the histogram. Once the probability is known, the state diversity level of that pixel is calculated by entropy equation given as equation 6.

$$E_{i,j} = -\sum_{q=1}^{Q} P_{i,j,q} \log\left(P_{i,j,q}\right) \quad (6)$$

In entropy based motion detection method, pixels at edges also get higher entropy, since both motion and spatial diversity can cause high entropy and they are hard to differentiate, which can lead to false detection of motion. To solve this problem, the spatial influence must be removed. One way to remove spatial influence is to calculate entropy based on difference image, i.e., the pixel's histogram is formed by accumulating pixels in difference images. Thus simple absolute image subtraction between consecutive frames removes the spatial influence, which can be processed as mentioned above to calculate the motion.

The following steps are used to detect moving object pixel based on difference based spatial temporal entropy. Calculation of difference images: here difference image is calculated by simple absolute difference of gray level as shown in equation 7. The difference image $D(k)$ is then quantized. Here we quantize the 256 gray levels into Q = 32 gray levels. Where, ψ represent quantization.

$$D(k) = \psi\left(\left|F(k) - F(k-1)\right|\right) \quad (7)$$

- Accumulation of pixel: to calculate entropy of each pixel in a particular frame, multiple difference images are needed to accumulate histograms from *D(k-L+1) to D(k),* where L is the number of frames for accumulation. To do motion detection accumulating process is computationally expensive. To solve this histogram $H_{i,j,q}$ for pixel *(i, j)* is updated online using simple recursive updates. Specifically, for the first L frames, from *D(1)* to *D(L)* histogram accumulation is given by equation 8 and for next consecutive frames histogram accumulation is given by equation 9. In this equation α is the weighting factor and it depends upon the intensity of motion.

$$H_{i,j,q}(k) = \frac{1}{L}\sum_{k=1}^{L} h_{i,j,q}(k) \tag{8}$$

$$H_{i,j,q}(k+1) = \alpha H_{i,j,q}(k) + (1-\alpha) h_{i,j,q}(k+1) \tag{9}$$

- Once the accumulated histogram of each pixel at the k^{th} frame is obtained, the PDF for pixel (i, j) is formed by normalizing the histogram which is given by equation 10.

$$P_{i,j,q} = \Gamma\left(H_{i,j,q}(k)\right) \tag{10}$$

Where, *Γ* denotes normalization i.e., the pixel number in each bin of $H_{i,j,q}$ is divided by the total number of pixels in the histogram. Since *D(k)* is quantized into *Q*=32 gray levels, the normalized histogram, $P_{i,j,q}$ has 32 discrete values. After probability is obtained entropy is calculated by equation 11.

$$E_{i,j}(k) = -\sum_{q=1}^{Q} P_{i,j,k}(k)\log\left[P_{i,j,k}(k)\right] \tag{11}$$

After obtaining entropy many methods can be used to derive the motion region. One simple method is to binaries the image by a threshold.

2.3. VOP Extraction

The different input frames in the sequence namely frame 1 and frame 20 are shown in the Figures 3 and 4. With the moving pixel obtained from spatial temporal entropy we extract the moving edge ME_n by edge change detection as illustrated from Figures 5 and 7. With a moving edge map, the VOPs are ready to be extracted. The horizontal candidates are declared to be the region inside the first and last edge points in each row and the vertical candidates for each column. After finding both horizontal and vertical VOP candidates, intersection regions through logical AND operation gives the pixels belonging to VOP, which are further processed by morphological operations. In this algorithm we used pixel connectivity to remove unwanted edge pixels other than moving edges. VOP is extracted by image filling using structure element corresponding to the extracted moving edge and the extracted video objects are shown in Figure 8. This VOP extraction can be carried out for any frame in the video sequence from the first to the last one.

Figure 3. Input image sequence frame 1.

Figure 4. Input image sequence frame 20.

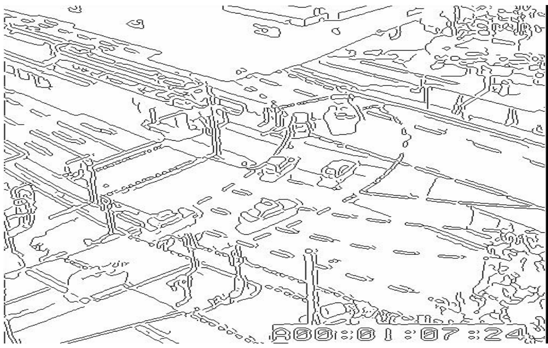

Figure 5. Edges of frame 1.

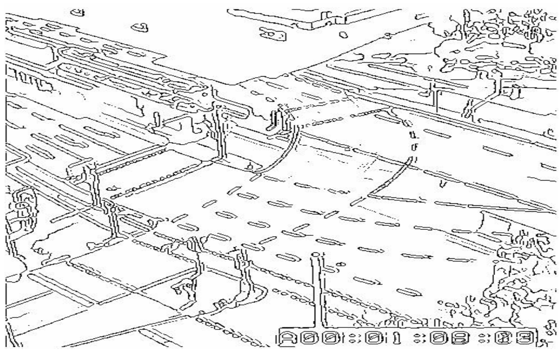

Figure 6. Background edges.

3. Model Matching Based Video Object Extraction

The algorithm implemented integrates tools from robust statistics and pattern recognition to automatically detect and extract moving objects in video sequences. The algorithm processes one video shot at a time and is composed of two stages: moving object detection and moving object extraction. The first stage is applied to the first two frames of a video shot to discover moving objects while the second stage is applied to the rest of the frames to extract the detected objects through the video shot.

Figure 7. Background subtracted moving edges.

Figure 8. Extracted video objects planes of frame 20.

The algorithm is applied to first two frames of the image sequence to detect the moving objects in the video sequence. First two frames of video sequence are taken and motion vectors are computed using Adaptive Rood Pattern Search (ARPS) algorithm as shown in Figure 9. Simultaneously, components of optical flow are computed for each block in the image. By using the motion vectors, motion compensated frame is generated which is given in Figure 10.

Figure 9. Motion vectors plotted.

Initial segmentation is performed on the first frame of traffic sequence. Applying watershed transformation directly on the gradient of image results in over segmentation as shown in Figure 11. To avoid over segmentation morphological gradient is computed on the frame and then watershed transformation is performed. After watershed transformation, some regions may need to be merged because of possible over-segmentation.

Canny binary edge image is used to localize an object in subsequent frames of video sequences and to detect the true weak edges. Intensity edge pixels are used as feature points due to the key role that edges play in the human visual process and the fact that edges are little affected by variation of luminance. Object models evolve from one frame to the next, capturing the changes in the shape of objects as they move. The algorithm naturally establishes the temporal correspondence of objects throughout the video sequence, and the output of the algorithm is a sequence of binary models representing the motion and shape changes of the objects.

Object model is obtained by subtracting background edge from edge image and eliminating unlinked pixels. After a binary model for the object of interest has been derived the motion vectors generated from ARPS algorithm are used to match the subsequent frames in the sequence. Matching is performed on edge images because it is computationally efficient and fairly insensitive to changes in illumination. The degree of change in the shape of an object from one frame to the next is determined based on simplified Hausdorff distance where simplified Hausdorff distance is defined as combination of distance transformation and correlation. Distance Transform the image and then threshold it by different amounts to form different dilated image sets. To search for the object in the image, it is required to obtain the amount by which the image is dilated such that maximum points in the object model are matched to image set.

Figure 10. Motion compensated image.

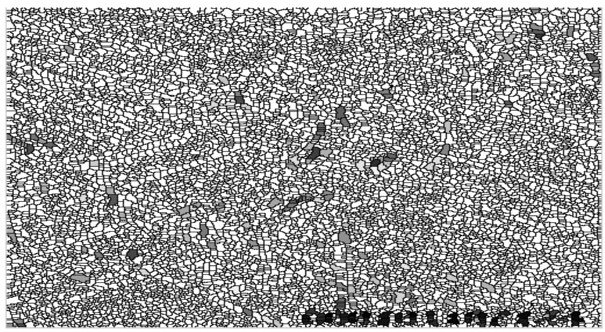

Figure 11. Over-segmentation result using watershed segmentation.

3.1. Motion Estimation

Motion estimation is based on temporal changes in image intensities. The underlying supposition behind motion estimation is that the patterns corresponding to objects and background in a frame of video sequence move within the frame to form corresponding objects on the subsequent frame. Motion estimation is accomplished using ARPS algorithm [12]. The ARPS algorithm makes use of the fact that the general motion in a frame is usually coherent, i.e. if the macro blocks around the current macro block move in a particular direction then there is a high probability that the current macro block will also have a similar motion vector. This algorithm uses the motion vector of the macro block to its immediate left to predict its own motion vector. The ARPS algorithm tries to achieve the Peak Signal to Noise Ratio (PSNR) similar to that of Exhaustive Search (ES) algorithm. ES algorithm, also known as Full Search, is the most computationally expensive block matching algorithm. This algorithm calculates the cost function at each possible location in the search window. As a result it finds the best possible match and gives the highest PSNR among any block matching algorithm. The obvious disadvantage of exhaustive search is it takes more computations to estimate motion vectors. ARPS tries to achieve the same PSNR as that of ES with less number of computations.

3.2. Initial Segmentation

A watershed-based algorithm presented is used for spatially segmenting the motion compensated image which is computed using motion vectors of video sequence. Watershed transformation [6] and [9] is known to be a very fast algorithm for spatial segmentation of images. It is performed on the gradient of the image to be segmented. Each minimum of the gradient leads to a region in the resulting segmentation. Conventional gradient operators generally produce many local minima which are caused by noise or quantization error. Hence, watershed transformation with a conventional gradient operator usually results in over segmentation. To alleviate this problem, the multistage morphological gradient operator is defined, which effectively enhances blurred edges and reduces the number of local minima. The watershed-based algorithm [9] is performed in the following five steps:

- Pre-filtering: in order to reduce noise disturbance, the original image to be segmented is pre-filtered using open–closing by reconstruction with a structuring element of size 3.
- Multistage Morphological Gradient: let $f(x, y)$ denote the pre-filtered image, and Bi a group of structuring elements. The size of B_i is $(2i+1)$ for $0 \leq i \leq 3$. The symbols $\oplus$ and Θ denote dilation and erosion, respectively. The multistage morphological gradient is defined as

$$MG(f) = \sum_{i=1}^{3} \left[((f \oplus B_i) - (f \Theta B_i)) \right] \qquad (12)$$

- Elimination of Small Local Minima: small local minima refer to local minima consisting of fewer than 2x 2 pixels or having a contrast lower than a constant. This kind of local minima in *MG (f)* is generally caused by noise or quantization error. They are removed first by dilating *(MG (f))* with a square structuring element of *2x 2* pixels. Then, the constant is added to the dilated gradient image. The final gradient image is obtained using reconstruction by erosion from $(MG(f) \oplus Bs) + H$. The constant is used to control the resulting number of segmentation regions. As H increases, the number of regions produced decreases.
- Watershed Transformation: watershed transformation is performed on the gradient image after the elimination of small local minima. The over- segmentation results obtained is shown in Figure 11.

3.3. Motion Based Region Merging

After watershed transformation also some regions have to be merged because of possible over-segmentation. Adjacent regions with coherent motion should be merged together to form a moving object. The similarity between two regions in terms of motion is usually measured by the Sum Squared Error (SSE). Regions are considered for merging by comparing their joint SSE.

$$E_{(r,f)}(u,v) = \sum_{(x,y)\in \text{Region}} [I(x,y) - I_f(x+u, y+v)]^2 \qquad (13)$$

Two neighboring regions (r_i and r_j) are assumed to have similar motion (and can thus be merged) if they satisfy the following equations:

$$\min(Er_i, fp) \leq \min(Er_i, fq) \qquad (14)$$

$$\min(Er_j, fp) \leq \min(Er_j, fq) \qquad (15)$$

$$min(Er_i, fp + Er_j, fp) = min(Er_j, fp) + min(Er_j, fp) \qquad (16)$$

In order to satisfy these three equations the reference frame f_p must result in the minimum motion compensation error for both the neighboring regions with respect to frame f_q. The additional error resulting from merging these two regions must be zero. Together, these three equations imply that the translational motion vectors of the joint region and the two individual regions will be identical. The results of the region merging process are shown in Figure 12.

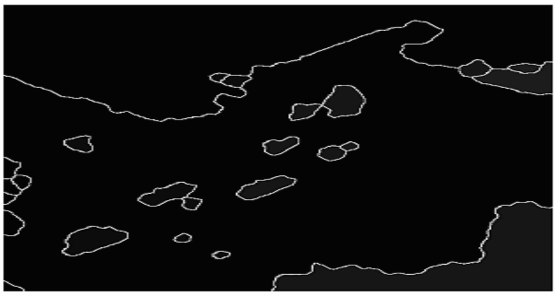

Figure 12. Region merging results.

3.4. Model Initialization

Object model is obtained by subtracting background edge from edge image and eliminating unlinked pixels. The Canny method [3] finds edges by looking for local maxima of the gradient of *I*. The gradient is calculated using the derivative of a Gaussian filter. The method uses two thresholds, to detect strong and weak edges, and includes the weak edges in the output only if they are connected to strong edges. This method is therefore less likely to be affected by noise, and more likely to detect true weak edges. The Canny based edge detection procedure involves the following steps:

- Smooth the image with a Gaussian filter,
- Compute the gradient magnitude and orientation using finite-difference approximations for the partial derivatives,
- Apply non-maxima suppression, with the aid of the orientation image, to thin the gradient-magnitude edge image,
- Track along edges starting from any point exceeding a higher threshold as long as the edge point exceeds the lower threshold.
- Apply edge linking to fill small gaps.

3.4.1. Background Edge Detection

Background edge map [4] contains absolute background edges in case of a still camera and can be extracted from the first frame or by counting the number of background edge occurrence for each pixel through the first several frames. In the present work reported here, we extracted background edge map from the first several frames by simple threshold. Pixel frequency is calculated by counting edge occurrence from complete frames. From the frequency value the background pixels are determined. Threshold value in decision is set to 50%. The background and foreground edges and the detected objects are shown in Figures 13, 12, 13, 14, and 15.

The Algorithm:

- Edges are obtained for all the frames in the traffic sequence.
- Counting edge occurrence from all the frames.
- Taking threshold as 50% of the count.
- The pixel values greater than or equal to the threshold hold value are set to 1 and others are set to 0.

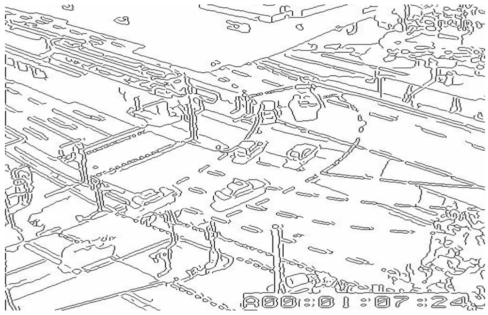

Figure 13. Background and foreground edges.

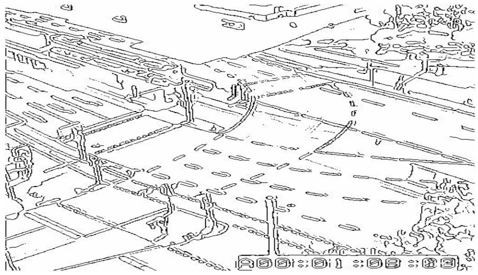

Figure 14. Background edges.

Figure 15. Detected objects.

3.5. Model Matching

The core of the technique is the object model matching that establishes temporal correspondence of objects throughout the video sequence. Object model is obtained by subtracting background edge from edge image and eliminating unlinked pixels. After a binary model for the object of interest has been derived, the motion vectors generated from ARPS algorithm [12] are used to match the subsequent frames in the sequence. Matching is performed on edge images because it is computationally efficient and fairly insensitive to changes in illumination. The matching results are shown in the Figure 17. The degree of change in the shape of an object from one frame to the next is determined based on simplified Hausdorff distance where simplified Hausdorff distance is defined as combination of distance transformation and correlation. Distance Transform the image and then threshold it by different amounts to form different dilated image sets as shown in Figure 16. To search for the object in the image, it is required to obtain the amount by which the image is dilated such that maximum points in the object model are matched to image set.

The simplified Hausdorff distance [7] is computed using the distance transformation, which determines for each pixel in the edge image. There are several approaches to the computation of the distance transform of an image. One commonly used approach is the Chamfer method. The Chamfer method defines small masks containing integer approximation of the Euclidean distances in a small neighborhood. Two such masks were suggested; Chamfer 3-4 and Chamfer 5-7-11, with the latter used in the present work due to its higher accuracy. The method involves initializing the distance transform to zero at every pixel in the binary image with a value of 1 and to infinity at every pixel with a value of 0. Then, the Chamfer mask is placed at every pixel to compute the distance transform and the process is repeated until no distance transform values are changed. To calculate Hausdorff distance, the Chamfer 5-7-11 mask is applied to approximate the distance.

Figure16. Distance transformed image.

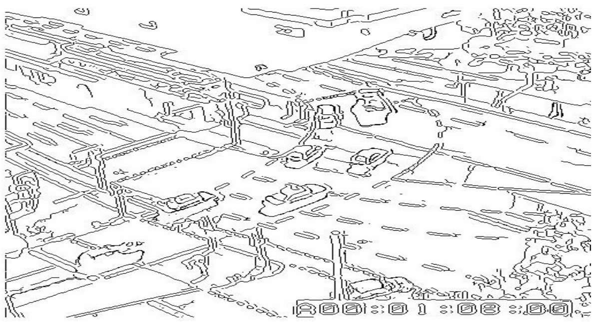

Figure17. Matching results on frame 2.

3.6. VOP Extraction Based on Model Matching

Let I_t and I_{t+1} denote two adjacent frames of a video sequence at times t and t+1, and let O_t and R_t and represent the object model and the object region at time t, respectively. If a best match translation $T=(x_t,y_t)$ is found for O_t in frame I_{t+1}, the difference signal $D=\{d(x,y)\}$ is given by

$$d(x,y)=\begin{cases}\left|I_t(x,y)\oplus T-I_{t+1}(x,y)\right| & (x,y)\in R_t\oplus T\\ +\infty & (x,y)\notin R_t\oplus T\end{cases} \quad (17)$$

Applying a threshold δ_R to *D*, a binary image D' is obtained where the regions with value “1” indicate the reliable part of the object and are used as the object marker. A morphological open operation is applied to D' in order to remove any possible noise before extracting the object marker. The uncertain area that may include part of the object can be determined by dilating the old object region by a disk of radius, shifting it to the best matching position, and then computing the logical AND of I_{t+1} with the dilated and translated object. The value of T reflects the degree of change in the shape of an object from one frame to the next. The background marker is extracted as the region of the image that does not include the uncertainty area. The reliable objects are extracted using markers. More accurate video objects are extracted than the already discussed method starting from the first to the end frames.

Figure18. More accurately extracted video objects.

4. Conclusion

The results obtained by VOP extraction based on edge change detection suffer from boundary inaccuracy. Change detection methods do not support moving background. As the number of objects in the image increases, boundary inaccuracy increases in case of change detection methods. Model matching techniques partially overcomes the drawbacks of change detection methods. Compared to the change detection methods, model matching method allows a moving background as well a stationary background. It also improves the segmentation results by distinguishing the overlapping of moving objects in the situation of multiple moving objects. As a new scheme for model update introduced to accommodate objects undergoing a complex motion and/or relatively large changes in shape, the proposed method generates more accurate object boundaries and is more robust in handling various real-world situations, including fast and/or intermittent motions, multiple moving objects, and partial occlusion. Unlike other VOP segmentation approaches, the proposed method not only generates VOPs but also provides object trajectories and evolving binary object models. Such features can be readily integrated into efficient indexing to facilitate content-based retrieval in video databases. It is also a fast algorithm to extract moving objects. The main difficulty is to obtain an initial model and to update it in the presence of cluttered background. Combining the motion vectors information and Hausdorff distance measure would significantly improve the quality of segmentation.

References

[1] Bouthemy P. and Francois E., "Motion Segmentation and Qualitative Dynamic Scene Analysis from an Image Sequence," *International Journal of Computer Vision*, vol. 10, no. 2, pp. 157-182, 1993.

[2] Camillo G., Octavia C., and Mario S., "Segmentation for Robust Tracking in the Presence of Severe Occlusion," *Computer Journal of IEEE Transaction on Image Processing*, vol. 13, no. 2, pp. 299-307, 2004.

[3] Canny J., "A Computational Approach to Edge Detection," *Computer Journal of IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 8 no. 6, pp. 679-698, 1986.

[4] Changick K. and Jenq H., "Fast and Automatic Video Object Segmentation and Tracking for Content Based Applications," *Computer Journal of IEEE Transactions on Circuits and System for Video Technology*, vol. 12, no. 2, pp. 122, 2002.

[5] Changick K. and Jenq H., "Fast and Robust Video Object Segmentation in Video Sequence," *Computer Journal of IEEE Transactions on Circuits and System for Video Technology*, vol. 11, no. 11, pp. 1160-1170, 2002.

[6] Demin W., "Unsupervised Video Segmentation Based on Watersheds and Temporal Tracking," *Computer Journal of IEEE Transaction on Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 525-538, 1998.

[7] Haifeng X. and Akmal A., "Automatic Moving Object Extraction for Content-Based Applications," *Computer Journal of IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 6, pp. 914-920, 2004.

[8] Jinhui P., Shipeng L., and Ya Q., "Automatic Extraction of Moving Objects Using Multiple Features and Multiple Frames," *Computer Journal of IEEE International Symposium on Circuits and Systems*, vol. 1, no. 1, pp. 413-416, 2000.

[9] Luc V. and Piene S., "Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations," *Computer Journal of IEEE Transactions of Pattern Analysis and Machine Intelligence,* vol. 13, no. 6, pp. 583-598, 1991.

[10] Meyer F. and Beucher S., "Morphological Segmentation", Communications Image Representation, vol. 1, no. 1, pp. 21-46, 1990.

[11] Moscheni F., Bhattacharjee S., and Kunt M., "Spatiotemporal Segmentation Based on Region Merging," *Computer Journal of IEEE Transactions Pattern Anal Machine Intelligence*, vol. 20, no. 9, pp. 897-915, 1998.

[12] Yao N. and Kai M., "Adaptive Rood Pattern Search for Fast Block Matching Motion Estimation," *Computer Journal of IEEE Transactions Image Processing*, vol. 11, no. 12, pp. 1442-1448, 2002.

Kavitha Ganesan graduated in electronics and communication engineering in 2001 from Joseph's College of Engineering, University of Madras, India. She earned her Master's degree in applied electronics in January 2003 from Anna University, Chennai, India.

Shanmugam Jalla obtained his ME and PhD degrees from MIT, Anna University, India. in instrumentation engineering. He served in the Indian Air Force for about 15 years and then joined the Department of Instrumentation Technology, MIT, Anna University, as lecturer since March 1984.