IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 17, Issue 3, Ver. II (May – Jun. 2015), PP 44-49
www.iosrjournals.org

Motion-Based Foreground Image Segmentation

Vinayak G Birdar
Computer Sciemnce And Engineering Bkit Bhalki 585328

Abstract: *A variety of useful applications demonstrate the need for precise motion-based segmentation of image data. Video compression techniques rely heavily on accurate and efficient representation of spatio-temporal regions possessing similar motion models. In addition a variety of cinematographic effects are potentially possible with successful separation of a moving object's image from an arbitrary background. Computer vision systems often depend on the ability to distinguish or describe a moving object in an image sequence.*

This paper proposes a methodology, which separates a moving foreground from a stationary background in a general video sequence. It replaces or removes moving persons or objects in motion pictures. The problem is motivated primarily by the need for replacing or removing moving persons or objects in motion pictures. The main complication exists in this problem is the issue of occlusions with the background. Consider a speaker that moves their hand behind a stationary background object such as a desk or chair. Such occlusions represent a challenge and are not examined in this approach. In addition to ignoring occlusions, the following analysis makes other assumptions about a given image sequence. Lighting and color changes should not be present to a high degree in the video being examined. Furthermore, the moving object should be constrained towards the interior of the image.

Keywords: *Image, image segmentation, motion estimator, noise estimator, recursive tracer, Block -matching algorithm.*

I. Introduction

Due to rapid progress in microelectronics and computer technology together with the creation of networks operating with various channels capacities, the last decade has seen the emerging of new multimedia applications such as Internet multimedia, video on demand (VOD), interpersonal communications (Video Conference, Video phone) & Digital Library. The importance of visual communications has increased tremendously. In addition however, motion-based segmentation also finds a niche in the realms of the video compression and computer vision.

In the analysis of objects in images it is essential that we can distinguish between the objects of interest and "the rest". The later group is also referred to as background. The techniques that are used to find the objects of interest are usually referred to as segmentation technique i.e. segmenting the foreground from background. Motion is powerful cue used by human being and animals to extract object of interest from background of irreverent detail. In imaging applications motion arises from a relative displacement between the sensing system and scene being viewed such as in robotic application, autonomous navigation and dynamic scene analysis. The task of image segmentation is to classify pixels as belonging to foreground or background objects. A common approach is to use pixel threshold, which make decision based on local pixel information that can be effective when pixel intensities are clearly above or below threshold. Thresholding produces a segmentation that yields all pixels that are belongs to the objects or objects of interest in an image.

The system is consisting of noise estimator, motion estimator, and recursive motion tracer. Selection of a motion estimator model represents the first step in the problem. Regardless of motion estimator, careful attention must be paid to noise effects when estimating motion. Faulty motion vectors due to image noise can lead to visually unpleasant effects such as isolated background blocks in the resulting segmented image. Noise-reduction filters may be used to alleviate this problem. Another method is to examine the resulting mean-squared error of known zero-motion vector regions. Any error must be due solely to noise and thus this provides information about the noise in a particular image sequence.

Accurate knowledge of all the motion vectors in a sequence theoretically provides the means to segment the images into pixels associated with a moving object and pixels associated with a rigid background. The algorithm for tracing motion vectors throughout the sequences is highly recursive and can be computationally expensive, depending on the number of non-zero motion vectors present. A video scene to be segmented should be motion traced both forwards and backwards temporally.

II. Objective Of The Study

The objective of this work is to develop an algorithm that separates moving foreground from a stationary background in a general video sequence. This problem is motivated primarily by the need for replacing or removing a moving object in a motion picture. As mentioned above, however, motion-based segmentation also finds a niche in the realms of video compression and computer vision. Selection of a motion estimator model represents the first step in the problem. Gradient-based methods such as optical flow have shown high performance but generally come with increased computational overhead than block-based matching. The disadvantage of block methods is an expected loss of sharpness at edge regions marking the boundary between foreground and background.

Regardless of motion estimator, careful attention must be paid to noise effects when estimating motion. Faulty motion vectors due to image noise can lead to visually unpleasant effects such as isolated background blocks in the resulting segmented image. Noise-reduction filters may be used to alleviate this problem. Another method is to examine the resulting mean-squared error of known zero-motion vector regions. Any error must be due solely to noise and thus this provides information about the noise in a particular image sequence. Accurate knowledge of all the motion vectors in a sequence theoretically provides the means to segment the images into pixels associated with a moving object and pixels associated with a rigid background. The algorithm for tracing motion vectors throughout the sequences is highly recursive and can be computationally expensive, depending on the number of non-zero motion vectors present.

A video scene to be segmented should be motion traced both forwards and backwards temporally. An interesting and difficult complication to this problem is the issue of occlusions with the background.

III. Implementation

The creation of segmentation system is accomplished in three stages as shown in fig 1.

Noise estimation

Moving vector detection

Tracing the moving vector

During the first stage the noise parameter are computed using least mean square error algorithm the motion vector between a pair of successive frames are computed without considering noise. The mean–squared error of regions that have known zero motion vectors must be solely a result of noise in the image sequence. The mean and standard deviation of the resulting collection of mean-squared error is computed and some number of standard deviation above the mean represent a threshold value. Using block-matching method carries the process of moving vector detection for motion parameter. Third stage function is to distinguish regions that are moving in any frame. Tracing motion vectors lends itself naturally to a recursive solution.

3.1 Noise Estimator

Before the start of motion estimation, an accurate estimation of the mean-squared error due to nose in pixel intensities must be obtained [7]. With this information, incorrect non-zero motion vectors may be discarded before they are traced. Based on the tracing algorithm used, the rejection of these vectors can have important consequences for the resulting visual quality. The noise estimation is computed in a straightforward way. Vectors between a pair of successive frames are computed without considering noise. Any mean squared error of regions along with non-zero motion vectors is a result of noise in the image sequence. The mean and standard deviation of the resulting collection of mean-squared errors is computed, and some number of standard deviations above the mean represents a threshold value.

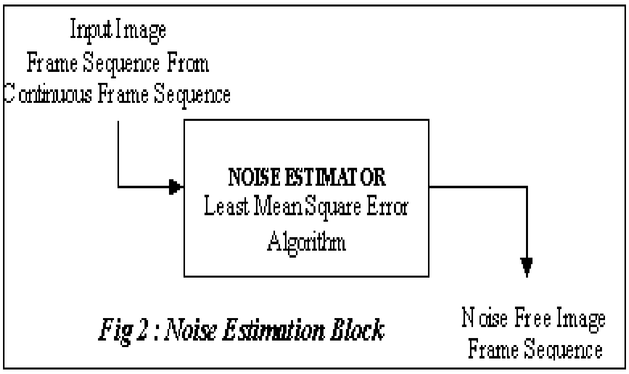

Fig 2 : Noise Estimation Block

In the calculation of motion vector, only minimum mean-squared errors that fall below this threshold can be considered true non-zero motion vectors. This method relies on the assumption that a large number of zero-motion regions can be found before considering noise. Sometimes the images are noisy; in that case the collection of data points for threshold estimation is small.

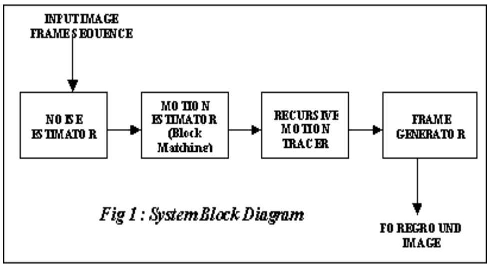

Fig 1 : System Block Diagram

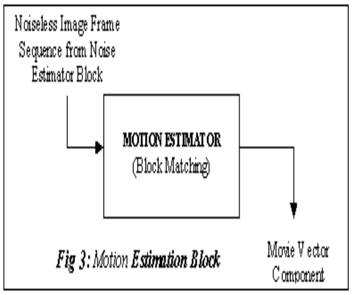

Fig 3: Motion Estimation Block

3.2 Motion Estimation

A block-matching motion estimator is used to calculate motion vectors over each pair of images in the sequence [1, 5]. Since high accuracy in the motion vectors is desired, the estimator performs a full-search over the window. The minimum mean-squared error criterion gives the best block match. The choice of the block size will have a great impact on results. A smaller block size will tend to produce more false motion vectors despite any noise estimation, but will result in finer edge definition in the resulting segmented image. Larger block sizes have coarser edges but are less plagued by noise effects. Furthermore the number of computations necessary for motion tracing goes down as the block size increases since there are fewer motion vectors to analyze. For the images analyzed (144 x 176 pixels) a block size of 8 pixels is used and seems to be a reasonable choice. Calculate Motion vectors over each pair of frames in sequence. The estimator performs a full search over the window.

Block-Matching Procedure

The basic idea behind block–matching procedure is depicted in the figure-4 where the displacement for a pixel (n1,n2) in frame k (present frame) is determined by considering an N1XN2 block centered about a pixel (n1,n2) and searching frame k+1 for the location of best matching block of same size. The process differ in (1) Matching criteria (2) Search strategy (3) Determination of block size. Uses Maximum Matching pixel count (MPC). Block size is calculated by doing continuous testing and taking combination of frame sizes with different frame skips. The matching of the blocks can be quantified according to various criteria including the maximum cross-correlation, the minimum mean square error (MMES), the minimum mean absolute difference (MAD) and maximum matching pel count (MPC).

Block Size: The selection of appropriate block is essential for any block based motion estimation. There are conflicting requirements on the size of the search block. If the blocks are too small, a match may be established between block containing similar gray level patterns, which are unrelated in the sense of motion. On the other hand, if the blocks are too large, then actual motion vectors may vary within a block, violating the assumption of a single motion vector per block. The block size for the proposed design is calculated by performing continuous testing and taking different combination of frame sizes with different frame skips.

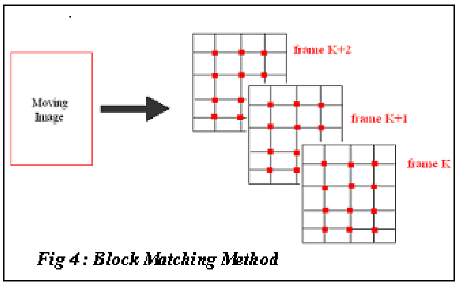

Fig 4 : Block Matching Method

3.3 Motion Tracer

This block represents the heart of the segmentation process [11] as shown in figure-5. Its function is to distinguish regions that are moving in any frame or have moved at any time throughout the sequence of images. For instance, in a complicated moving object such as a person, a hand or shoulder may only move slightly between a single pair of frames and will be stationary in all other frames. Ideally the tracer recognizes this and segments the region over all the frames, not just the frames in which it moved. Tracing motion vectors lends itself naturally to a recursive solution. Each block with non-zero motion vectors in each frame represents a "seed" call to the tracing function.

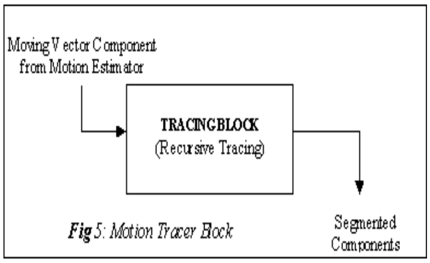

Fig 5: Motion Tracer Block

The figure-6 illustrates a moving block, which translates into a region corresponding to four blocks. The tracing process begins with a seed call where the seed block will move into as many as four other blocks, and each of these blocks is recursively called by the tracing function. The purpose of the tracing function is simply to identify the appropriate moving pixels based on the motion vectors and block regions. Motion tracing has a straightforward solution only in a one direction temporally. In other words, tracing must be done in both the forwards and reverse temporal directions for best segmentation results.

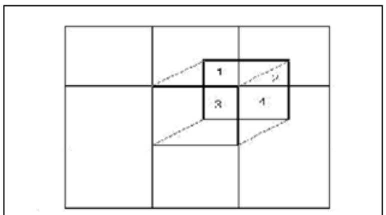

Fig 6 : General Motion seed block into 4 new block

Figure-7 shows three varieties of the tracing process. The first case represents the fully general approach. For any moving block only the pixels corresponding to that moving block are associated with motion, but all four regions impinged by the block are seeded to the successive tracing call. This is the most accurate approach but also the most computationally burdensome. The second approach is to seed all four blocks as well, but to treat all pixels within the four seeded blocks as having moved rather than just the actual moving pixels itself. A final approach is to mark all moving pixels as in the general case, but to only seed the block corresponding to maximum overlap. If there are equal overlaps, then multiple blocks are seeded. Although this variation only approximates the tracing problem, it can be much faster since each trace call usually only seeds one recursive call rather than four. In the most general case, the tracing algorithm runs slow. For improved speed, motion vectors are computed not between each frame but between every n frames and tracing is done on this smaller set of motion vectors. This explains the use of the optional interpolation estimator at the end of the block diagram in figure-1. This interpolation can improve temporal resolution that may be lost by skipping every n frames. The current implementation uses n=5 and does not use interpolation to increase temporal resolution

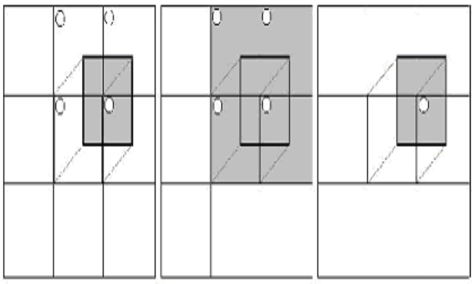

Figure 7: Varieties of Trace Algorithm Seeding and Pixel Mapping

IV. Experimental Results:

The figure-8 shows the input image frame Sequence from news.qcif, various image sequences are considered for regress testing of the implemented design, the image sequences are classified depending upon the frame sequence and images are chooses depending upon the level intensities and frame size and frame skips.

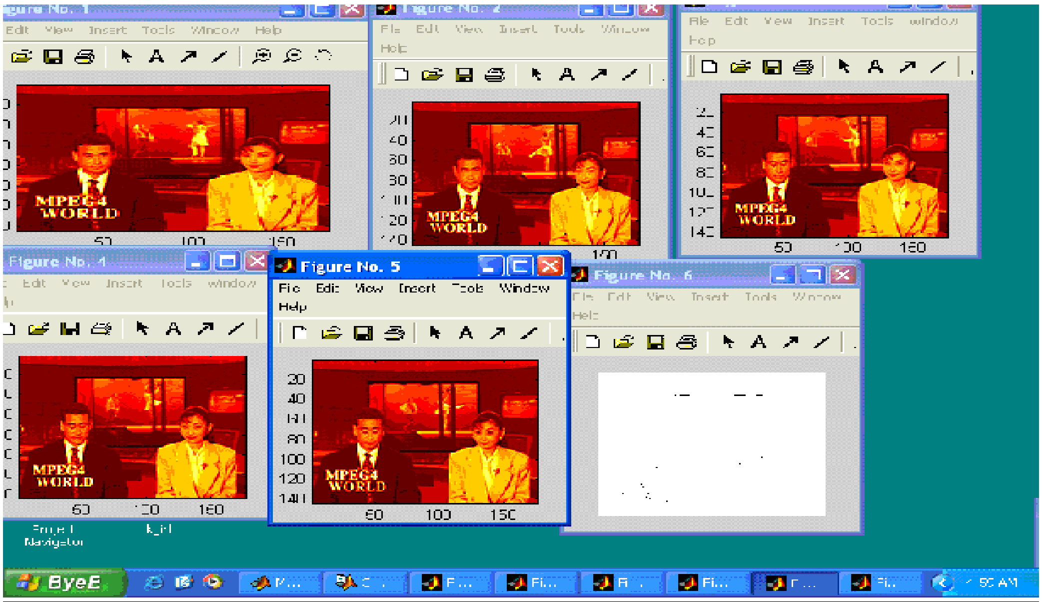

Fig 8 : Input Frame Sequence

Design is being tested for both gray level images as well as color image sequence. The figure-9 shows the segmentation of input frame sequence with number of frames #6 and frame skips #5.

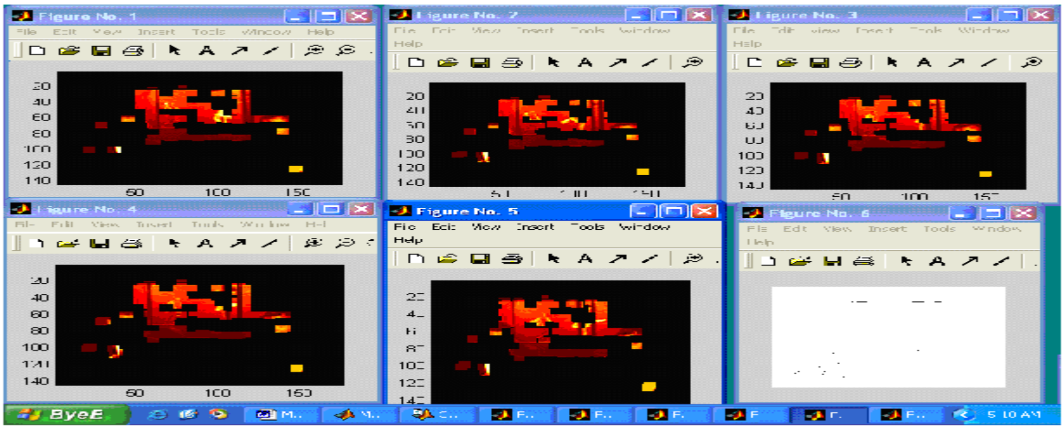

Fig 9 : Foreground Segmented Result

V. Conclusions

The methodology proposed here performs an efficient segmentation of moving foreground object from a given image sequence with a still background. From the result obtained it can be observed with the increase in frame size and keeping constant frame skip, the clarity of the segmentation is increased and results in higher resolution of segmentation because of the difference between two neighboring frame is quite less. A smaller threshold value in this case has not been found effective due to excessive loss of true motion information. Mean-squared error approach.

References

[1]. B. Han and L. Davis, On-Line Density-Based Appearance Modeling for Object Tracking, ICCV 2005.
[2]. Mittal and N. Paragios, Motion-based Background Substraction using Adaptive Kernel Density Estimation,CVPR, 2004
[3]. Ludwig Schwardt "The Least-Mean-Square Algorithm" University of Stellenbosch, 6th May 2004
[4]. P.R. Giaccone and G.A. Jones. "Segmentation of Global Motion using Temporal Probabilistic Classification." Proceedings of British Machine Vision Conference, pages 619-628, University of Southampton, 1998.
[5]. Wang and Adelson. "Spatio-Temporal Segmentation of Video Data." Proceedings of the SPIE: Image and Video Processing II, vol. 2182, San Jose, CA, February 1994
[6]. Wang and Adelson. "Representing Moving Images with Layers." IEEE Transactions on Image Processing, vol. 3, no. 5, September

1994.

[7]. Ioannis Kompatsiaris George Mantzaras and Michael G Strintzis "Spatiotemporal segmentation and tracking of objects in color image sequences" information processing laboratory Electrical and computer Engineering Department Aristotle University of Thessaloniki Greece.

[8]. Farneback G. "Motion Based Segmentation of Image Sequence", Linkoping University, SE-581 83 Linkoping, Sweden, 19996. LiTH-ISY-EX-1596.

[9]. Jnhui Pan Shipeng Li, Ya-Qin Zhang,"Automatic extraction of moving objects using multiple features and multiple frames" Tsinghua University Microsoft Research, China.

[10]. Ludwig Schwardt "The Least-Mean-Square Algorithm" University of Stellenbosch, 6th May 2004

[11]. Jong Bae Kim and Hang Joon Kim "Efficient Region-Based Motion Segmentation or a Video Monitoring system" Artificial Intelligence (AI) Research Group, Dept of Computer Engineering Kyungpook National University 1370 Sangyuk-Dong, Puk-gu, Daegu, 702-701, Korea.

[12]. Hwang-Seok Oh and Heung-Kyu Lee, Kajeong-dong, Yusong-gu, Taejon "Block-Matching Algorithm based on an adaptive reduction of the search area of the motion estimation" Virtual Reality Research and development center, ETRI, 161.

[13]. Y. Ted Wu, "Image Segmentation The first step in 3D imaging" Able software corporation,

[14]. Peter Dornbach "Implementation of Bi-directional ray tracing algorithm" Technical University of Budapest, Dept of Control Engineering & Information Technology, Hungery.

[15]. Mohamad Irfan Kazim " Block Matching techniques of Motion Estimation for object tracing"

[16]. R.C.Gonzalez, R.E.Woods " Digital Image Processing", second edition, Pearson Education 2002.