International Journal of Innovation and Scientific Research
ISSN 2351-8014 Vol. 30 No. 3 May 2017, pp. 377-384

http://www.ijisr.issr-journals.org/

Segmentation for Video Sequence

M. Ganeshwade Mandakini[1], Daivashala R. Deshmukh[2], and K. Vengatesan[3]

[1]Assistant Professor, Department of Computer Science Engineering, Marathwada Institute of Technology, Aurangabad, India

[2]Assistant Professor, Department of Computer Science Engineering, Maharashtra Institute of Technology, Aurangabad, India

[3]Assistant Professor, Department of Computer Science Engineering, Marathwada Institute of Technology, Aurangabad, India

ABSTRACT: The objective of this paper is generation of an algorithm that will separate moving foreground from a stationary background in a general video sequence. We use the different models to calculate the foreground motion in a robust estimation framework. Segmentation of objects in image sequence is very important in many aspects of multimedia applications. We describe a system for representing moving images from the multi-layered sequence. This work realizes a motion- based image isolation algorithms for isolating the moving image in a multi-layered moving image sequence. The system has been proposed here which can efficiently segment a moving foreground object from a given image sequence with still background. The modules of the system are developed using MATLAB and verified for its functionality. In our system different algorithms like LMSE, Block-matching algorithm, motion tracing and recursive algorithms are used to estimate foreground image segmentation for Multi-layered video sequence. Experimental results are given to show the efficiency of our methods.

KEYWORDS: Block-matching, frame-interpolator, image/video segmentation, motion estimator.

1 INTRODUCTION

Due to the rapid progress in microelectronics and computer technology together with the creation of networks operating with various channel capacities, the last decade has seen the emerging of new multimedia applications such as Internet multimedia, Video on Demand (VOD), interpersonal communications(videoconference, videophone), and digital library.

The content-based functionalities in MPEG-4 video characterize a revolution in representing digital video and will have a tremendous influence in the future of visual world. With content-based functionalities, video bit stream can be manipulated to achieve personalized video. In MPEG-4 video, bit stream is composed of Video Object Planes (VOP), which can be used to assemble a real world scene. Each VOP is coded independent of other objects by its texture, motion and shape. VOP form the basic elements of MPEG-4 video bit streams. VOP extraction is a key issue in efficiently applying the MPEG-4 coding scheme. Although MPEG-4 standard doesn't specify how to obtain VOP, it's apparent and recognizes that segmentation-based techniques are essential to, and therefore, will dominate VOP generation. This is because most of visual information, existing or being generated, is in the form of frames or images. To achieve content-based functionalities, these frames and images have to be decomposed into individual objects to be fed into MPEG-4 video encoder.

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.

1.1 SEGMENTATION

The ideal goal of segmentation is to identify the semantically meaningful components of an image and grouping the pixels belonging to such components. While it is impossible to segment static objects in image at the present stage, it is more practical to segment moving objects from dynamic scene with the aid of motion information contained in it. Segmentation of moving objects in image sequence plays an important role in image sequence processing and analysis. Once the moving objects are detected or extracted out, they can serve for varieties of purposes.

The aim of image segmentation is to segment the video frame into background, foreground, objects, and sub objects with different characteristics so that the mesh can represent the motion of objects perfectly. During segmentation, luminance, color and motion are used. In fact, decomposing a video sequence into Visual Object (VO) is very difficult. An intrinsic problem of VO generation is that objects of interest are not homogeneous with respect to low-level features such as color, intensity, or motion. Thus, it is very hard to obtain meaningful partitions

1.2 MULTI-LAYERED IMAGE

A system is described for representing moving images with sets of overlapping layers. Each layer contains an intensity map that defines the additive values of each pixel, along with an alpha map that's servers as a mask indicating the transparency. The layers of ordered in depth and they occlude each other in according with the rules of composting. Velocity maps define how the layers are to be warped over time. The layered representation is more flexible than standard image transforms and can capture many important properties of natural image sequences

The decomposition of a sequence into its layered image representation is presented and demonstrates the flexible of the layered decomposition with applications in image compression and video special effects. The general block diagram of the algorithm consists of motion estimation, motion segmentation, and temporal integration is also presented.

2 DESIGN IMPLEMENTATION

SYSTEM DESIGN

This project work realizes a motion based image isolation algorithm for isolating the moving image in a multi layered moving image sequence. The system reads a moving data and read the frame sequence based on the value passed by the user. The obtained frame sequence is processed using motion estimator and tracer module for the isolation of moving parameters in the given moving image sequence

2.1 FRAME READER

This unit is the first interfacing unit for the interface of selected moving image to the implemented design. The module reads the complete moving image data and creates multiple frames based on the frame skip and the frame number passed. This module creates frames at a rate of frame skip passed to and select the frames depending upon the total frame numbers passed. These frames are the passed to the next noise estimator module for the calculation of noise present in the frame sequence.

2.2 NOISE ESTIMATOR

Before motion estimation is begun, an accurate estimate of the mean-squared error due to noise in pixel intensities must be obtained. With this information, incorrect non-zero motion vectors may be discarded before they are traced. The rejection of these vectors can have important consequences for the resulting visual quality. The noise estimation is computed in a straightforward fashion where the inter-pixel differences between the blocks of two successive frames are computed. On every process the obtained inter-pixel difference is considered to be least value and compared with the next difference value. If the difference is found to be less than the present difference the minimum error is taken as the present difference else the minimum error is retained the previous value. This operation is iterated for the whole block and for the whole image. The obtained minimum error values are the squared and the mean square error is calculated from it.

The least mean square error is given by;

LMSE error = [mean(least error)^2) + 3* σ (least error)]

This module reads the frame sequence from the frame reader and process on successive frames returning the least MSE (Mean square error) for each frame. The obtained value is then used for the elimination of noise effect in the given frames sequence.

NOISE ESTIMATION MODULE

In all following motion vector calculations, only minimum mean-squared errors that fall below this threshold value 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. The problem here is that if the images are very noisy or a particularly noisy pair of images is used for the threshold estimate, then the collection of data points for threshold estimation is small. Also, depending on the nature of the distribution of the mean-squared errors, a varying number of standard deviations above the mean results in a useful threshold value.

2.3 MOTION ESTIMATOR

A block-matching motion estimator is used to calculate motion vectors over each pair of frames in the sequence. 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 (144x176 pixels) a block size of 8x8 pixels is used for the implementation.

The module reads the frames sequences with the threshold value obtained from the noise estimator unit and implements the block matching algorithm as explained in section for the

2.4 MOTION TRACER

This block trace out the recursive pixels occurring in different frames rather than the successive frames. Its function is to distinguish regions that are moving in any frame or have moved at any time throughout the sequence of images. By analyzing the motion in this way, segmentation of regions that only move briefly is possible. 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. Isolation of such slow moving elements may not be possible under motion estimator block as it estimates the moving parameters between two successive frames only. The tracing unit realized, takes the frame sequence and trace out the whole frames for a pixel position in forward direction i.e. From the first to the last frame of the moving frame sequence is traced out for each pixel. If the difference is found gradually varying then such pixel values are also treated as moving. The unit implements the tracing algorithm for the determination of motion elements using forward tracing algorithm as explained in section

2.5 FRAME INTERPOLATOR

This unit reads the isolated motion vectors from motion tracer as well as motion estimator unit and rebuilds the frames depending on the motion vectors obtained. This module place the pixels back to the same position as in the original frames for the pixels isolated. Remaining pixel positions are been filled with zeros to neglect the non-moving elements. This unit after reconstructing the frames arrange it in sequence and play back as moving image. The obtained moving image is the final reconstructed isolated moving image for the given image. This isolated image can be taken and used for various other applications.

3 ALGORITHM FOR MULTILAYERED VIDEO

The moving frames are generally represented as a sequence of multiple frames. These frames are static in nature when isolated. All these frames together create a moving image as shown in figure 3 On a closer observation it can be seen that most of the moving frames has got correlated pixels among successive frames transmission of these correlated pixels for low bit rate application is a very difficult task. To overcome this difficulty the moving image can be isolated from the stationary elements and can be isolately transmitted for more efficient low bit rate application.

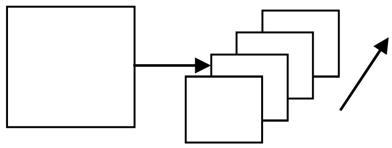


Fig 3 multi-frame representation of moving image

Generally images have two layers named foreground and background. Under moving image there are three probability occurs

1) Both foreground and background moving

2) Foreground stationary and background moving

3) Background stationary and foreground moving

In most of the moving images we find the probability of third observation occurring i.e. the Background stationary and foreground moving. During transmission if this foreground can be isolated and transmitted it can improve the operational efficiency of the system. This project work implements an advanced moving image isolation system using bock matching algorithm for foreground segmentation. The algorithms used are explained in the following sections.

LEAST MEAN SQUARE ERROR (LMSE) ALGORITHM

The Least Mean Square (LMS) algorithm is an adaptive algorithm, which uses a gradient-based method of steepest decent. LMS algorithm uses the estimates of the gradient vector from the available data. LMS incorporates an iterative procedure that makes successive corrections to the weight vector in the direction of the negative of the gradient vector, which eventually leads to the minimum mean square error. Compared to other algorithms LMS algorithm is relatively simple; it does not require correlation function calculation nor does it require matrix inversions.

3.1 BLOCK MATCHING ALGORITHM

The motion estimation and compensation technique has been widely used in video compression due to its capability of reducing the temporal redundancies between frames. Most of the algorithms developed for motion estimation so far are block-based techniques, called block-matching algorithm (BMA). In this technique, the current frame is divided into fixed size of blocks, & each block is compared with candidate blocks in reference frame within the search area. The widely used approach for the BMA is the full search BMA (FSBMA), which examines all candidate blocks within the search area in the reference frame to obtain a motion vector (MV). The MV is a displacement between the block in the current frame and the best matched block in reference frame in horizontal and vertical directions.

The motion estimation algorithm is performed with a variable size of search area depending on block types varying from 8x8 block to complete frame. The video sequences for low bit-rate video coding applications such as videophone and video-conferencing have some restrictive motion characteristics.

A block in a specific region in the previous frame can belong to the same region at that position in the current frame: a block in background region may lie in the background region in the current frame. The changing block shows the percentage of the difference from background to active region or vice versa. The other labels mean that the block types are the same in successive frames. In all video sequences, the percentage of background blocks in successive frames is very high.

The pattern of distribution is very similar without regard to video sequences. It is shown that the temporal correlation between successive frame is very high, that is, if a block in the previous frame belongs to background regions or active regions, the block which is located in the same position in the current frame may be classified as a background block or active moving block, respectively, with a strong probability.

The basic idea of block matching is depicted in the figure where the displacement for a pixel (n_1,n_2) in frame k (the present frame) is determined by considering an N1 x N2 block centered about (n_1, n_2) and searching frame k+1 (the search frame) for the location of the best-matching block of the same size. The search is usually limited to an N1+2M1 x N2+2M2 region called the search window for computational reasons.

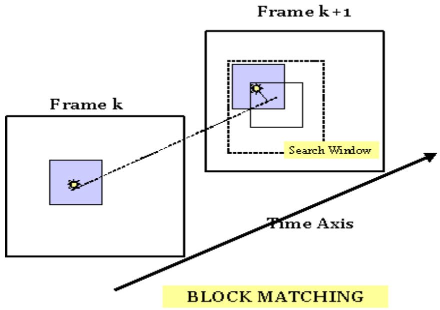


Figure 3.1 Block-matching

Block matching algorithms differ in

- The matching Criteria
- The search strategy
- The determination of block size

MATCHING CRITERIA

The matching of the blocks can be quantified according to various criteria including the maximum cross-correlation, the minimum mean square error (MSE), the minimum mean absolute difference (MAD) and maximum matching pel count (MPC)

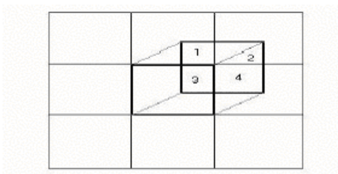


Figure 3.2 seed

MSE: $MSE(d_1,d_2) = 1/(N1\ N2)\ \Sigma\ [s(n_1,n_2,k) - s(n_1+d_1,n_2+d_2,k+1)]^2$

The estimate of motion vector is taken to be the value of (d1, d2) which minimizes MSE.

$[d_1, d_2]T = \arg.\min MSE(d_1, d_2)$

Normally not used, as it is difficult to realize the square operation in hardware

MAD: $MAD(d_1, d_2) = 1/(N1\ N2)\ \Sigma\ [|\ s(n_1, n_2, k) - s(n_1+d_1, n_2+d_2, k+1)\ |]$

$[d_1, d_2]T = \arg.\min MAD(d_1, d_2)$

Performance of MAD deteriorates as the search area becomes larger due to the presence of several local minima.

MPC: $T(n_1, n_2; d_1, d_2) = 1,$

if $|\ [s(n_1,n_2,k) - s(n_1+d_1,n_2+d_2,k+1)]\ | <= t$; 0, Otherwise

$MPC(d_1, d_2) = \Sigma\ T(n_1, n_2; d_1, d_2)$

$[d_1, d_2]T = \arg.\max MPC(d_1,d_2)$

The selection of an appropriate block is essential for any block-based motion estimation algorithm. There are conflicting requirements on the size of the search blocks. If the blocks are too small, a match may be established between blocks 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. Hierarchical block matching discusses these conflicting requirements. The block size for the proposed design is calculated by performing continuous testing, taking different combination of frame sizes with different frame skips.

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. As shown in figure 3.2 above, a moving block will, in general, translate into a region corresponding to four blocks. The tracing algorithm begins with a seed call. This 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, then to seed further calls to it. 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 and best segmentation results.

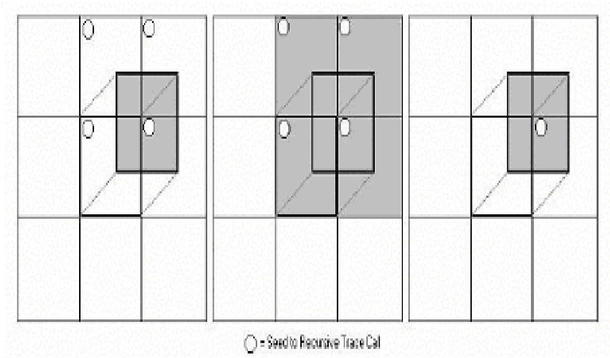


Figure 3.3 Tracing algorithm

Figure 3.3 shows three varieties of the tracing algorithm. 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 it. This approximation greatly simplifies the tracing algorithm, and also increases the algorithm efficiency dramatically, since a block that is seeded to the tracing function need not be ever seeded again

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. This interpolation can improve temporal resolution that may be lost by skipping every n.

MATLAB feature a family of application specific solutions called toolboxes. Toolboxes are comprehensive collection of MATLAB functions (M-file) that extends the MATLAB environment to solve particular classes of problems. Areas in which toolboxes are available include signal processing, image processing, control system, neural networks, fuzzy logic and many others. This high performance numerical computation package with built in function for graphics and animation. It can perform symbolic algebra like mathematical, Maple and Matrix X.

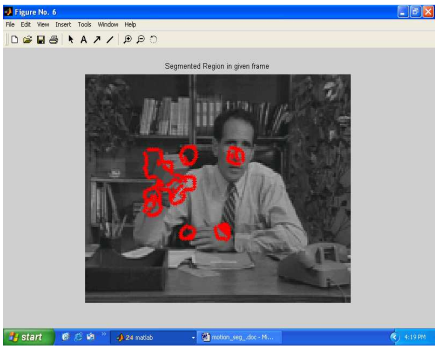
This allows solving many technical computing problems, especially those with matrix and vector formulations, in a fraction of the time it would take to write a program in a scalar non-interactive language such as C or FORTRAN. Environment to solve particular classes of problems. Areas in which toolboxes are available include signal processing, control systems, neural networks, fuzzy logic, wavelets, simulation, and many others.

EXPERIMENTAL RESULTS

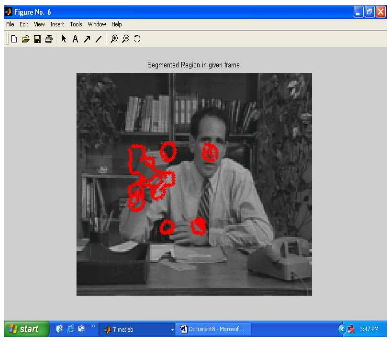
INPUT CONSIDERATION

Test input image frame sequence (Salesman. qcif) Fig given below shows that, input image frame sequence from salesman. qcif file. File has been loaded into current directory work library for simulation. Various image sequences are considered for regress testing of the implemented design. The image sequences are classified depending up on the frame sequence the moving sequence of the image. Images are furthered chooses depending upon the level intensities. Design is been tested for both gray level images as well as color image sequence

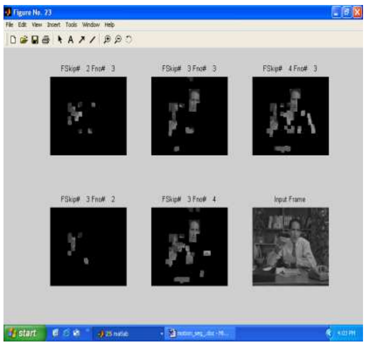
PERCENTAGE OF SEGMENTATION AT SKIP 2 &FRAMES3=9.13%)



PERCENTAGE OF SEGM ^{N} AT SKIP 3 FRAMES 3=(12.27%)



FIVEOBSERVATION PARAMETERS



A system has been proposed here which can efficiently segment for segmentation always remains a challenge as decrease in the frame skip value a result in high clarity of segmented image but increase the operation time, thus resulting in a slower system. On the other hand increase in frame skip and keeping constant frame size results in segmentation with less clarity. Though the speed of operation increases, because of higher value of frame skips, the two neighboring frame will be farther which when computed between the two consecutive frames gives very high difference value . Observing the other way where the frame skip is kept constant with the increase in frame size results in segmentation clarity.

4 CONCLUSION

The algorithm is decomposition in a way that will hopefully render future upgrades and modifications relatively simple. A particular weakness of the current approach is block artifacts near the foreground image edges. A different motion estimation algorithm such as optical flow may go a long way towards reducing these Undesirable effects. In addition, optical flow methods may show greater resistance to unwanted noisy motion vectors that debilitate the current approach. Noise-reducing filters may be an alternative method towards reducing noise-generated vectors, as opposed to the current threshold mean-squared error approach.

The current tracing algorithm, particularly the full four-block fill approximation, has shown good results towards the goal of segmentation. Again, with improved motion estimation techniques, the tracing algorithms may show improved results.

REFERENCES

[1] David B.Brown. "*Motion-Based Foreground Segmentation.*" digital video processing, EE392J Stanford University March 13,2000.

[2] P. R. Giaccone and G.A. Jones. "*Segmentation of Global Motion using Temporal Probabilistic Classification.*" Proceeding of British Machine Vision Conference, pages 619-628, University of Southampton, 1998.

[3] C.Nagarajan and M.Madheswaran – 'Analysis and Implementation of LLC-T Series Parallel Resonant Converter with Fuzzy controller'- International Journal of Engineering Science and Technology (IJEST), Applied Power Electronics and Intelligent Motion Control. Vol.2 (10), pp 35-43, December 2010

[4] C.Nagarajan and M.Madheswaran - 'Performance Analysis of LCL-T Resonant Converter with Fuzzy/PID Using State Space Analysis'- Springer, Electrical Engineering, Vol.93 (3), pp.167-178, September 2011.

[5] C.Nagarajan and M.Madheswaran - 'Experimental Study and steady state stability analysis of CLL-T Series Parallel Resonant Converter with Fuzzy controller using State Space Analysis'- Iranian Journal of Electrical & Electronic Engineering, Vol.8 (3), pp.259-267, September 2012.

[6] Wang and Adelson. "*Spatio- Temporal Segmentation of Video Data.*" Proceedings the SPIE: Image and Video .Processing II, vol. 2182, San Jose, CA, February 1994

[7] Wang and Adelson. "Representing Moving Images with Layers." IEEE Transactionson images Processing, vol. 3, no. 5,

[8] September 1994, revised May 1994

[9] Getting started with MATLAB By – Rudra Pratap , Oxford University Press 2003 . 6. Digital Image Processing . Using MATLAB – R. C. Gonzalez, third edition.

[10] MATLAB Programming for Engineers BY – Stephen J. chapman, Thomson learning Asia Pvt .Ltd, Singapore 2005, third edition.

[11] Vengatesan K., and S. Selvarajan: Improved T-Cluster based scheme for combination gene scale expression data. International Conference on Radar, Communication and Computing (ICRCC), pp. 131-136. IEEE (2012).