

# A DCT-Domain Video Alignment Technique for MPEG Sequences

Ming-Sui Lee, Meiyin Shen and C.-C. Jay Kuo  
Integrated Media Systems Center and Electrical Engineering  
University of Southern California, Los Angeles, USA  
E-mail: mingsuil@usc.edu, {meiyinsh, cckuo}@sipi.usc.edu

**Abstract**—An image/video registration technique for multiple compressed video inputs such as MPEG sequences is investigated. The proposed technique is based on the matching of discrete cosine transform (DCT) coefficients and motion vectors. First, the I frame of each input sequence is separated into the background and moving objects. For the background, coarse edge features are extracted by applying edge detectors of different characteristics to the luminance DC coefficients. Each detector generates a difference map for a single background. A threshold is determined for each difference map to produce a binary map. Then, alignment parameters are determined using the binary maps of input images generated by the same detector. For the moving object, alignment parameters can be finetuned by the motion information of all frames in the same group of pictures (GOP). Finally, the actual displacement in the pixel domain is estimated by the weighted average of alignment parameters from all background detectors and refinement parameters from motion information. It is shown by experimental results that the proposed method reduces the computational cost of image/video registration significantly in comparison with the traditional pixel domain registration techniques while achieving certain quality of composition.

## I. INTRODUCTION

Image/video registration is the process of aligning two or more images/videos taken by different cameras from different viewpoints. Applications of image/video registration techniques can be found in computer vision, pattern recognition and remotely sensed data processing. This topic has been studied for several decades, but most techniques were primarily developed based on the information of the pixel domain (or the spatial domain), which will involve tedious inverse/forward DCT when the input video is of the compressed format such as motion JPEG and MPEG. They are not suitable for real-time applications due to the heavy computation demanded. In this work, we study image/video registration techniques based on multiple captured and motion compensated compressed video clips, where DCT provides a powerful tool for energy compaction so as to remove spatial redundancy of the underlying image while motion vectors give auxiliary temporal information for alignment. To speed up the registration process, it is preferable to perform the registration task directly in the DCT domain to avoid the pixel-DCT domain conversion.

The main advantage of image/video registration in the DCT domain is that the computational complexity can be significantly reduced. However, this is by no means a straightforward

job since image registration is inherently developed in the spatial domain. It demands some careful thought in the design of a DCT-based registration technique. In our proposed method, coarse edge features are extracted out by applying several edge detectors of different geometrical meanings to the DC values of the luminance (Y) component. Then, a threshold is used to filter out minor features to generate binarized images. Finally, the estimation of the displacement parameter is conducted based on these binarized images. Since the proposed method is a block-based approach, the alignment accuracy is further improved by additional information such as motion vectors. Based on the moving object, alignment parameters can be finetuned by the motion information of all frames in the same group of pictures (GOP). Finally, the actual displacement in the pixel domain is estimated by the weighted average of alignment parameters from all background detectors and refinement parameters from motion information. It is shown by experimental results that the proposed method reduces the computational cost of image/video registration significantly in comparison with the traditional pixel domain registration techniques while achieving certain quality of composition.

This paper is organized as follows. A brief review of previous work is given in Sec. 2. The proposed algorithm is described in detail in Sec. 3. Experimental results and their discussion are given in Sec. 4. Finally, concluding remarks are given in Sec. 5.

## II. REVIEW OF PREVIOUS WORK AND RESEARCH MOTIVATION

Image/video mosaic, which combines several image/video inputs into a panorama output, has been widely studied. When the input image/video contents are taken from different viewpoints, sampling times and sensors, image registration is demanded to integrate these image/video tiles together. Over the past few decades, a large amount of work has been conducted to obtain image/video mosaic. For an extensive survey of previous work, we refer to [1] [5].

Generally speaking, the image/video registration technique consists of two major steps: feature detection and feature matching. Feature detection can be done either manually or automatically. Although it is straightforward for people to choose the matched patterns, it is desirable to develop an automatic feature selection process based on the particular application context for computer processing. Feature detection

techniques can be classified into two categories: the feature-based and the area-based approaches. The main task of the feature-based approach is to extract salient points such as corners, line intersections, line ends and centroids of closed-boundary regions [2], [3]. The area-based approach uses the correlation function to determine the degree of closeness. To be more specific, it computes the cross-correlation of intensities of a certain region of input images to find the best match. Once the feature information is available, the next step is to find the optimal correspondence between image tiles. Feature matching is a process to determine the relationship between similar objects contained in different images by finding spatial relations among extracted features. There is another work which consists of a hierarchical scheme that uses both spatial and temporal information for alignment [4].

Although the above methods lead to good results, some of them require specific instruments and they were primarily developed in the pixel (or spatial) domain, which is not applicable to general consumers and computationally expensive. Since Motion JPEG, MPEG and H26x coding standards all adopt the DCT representation during the coding process, given multiple compressed video inputs, it is desirable to conduct the registration process directly in the DCT domain to generate the corresponding compressed image/video mosaic. For a more generic scenario, we may consider multiple video sources captured by an arbitrary number of cameras with different parameter settings. There arise many challenging problems in creating video mosaic, including temporal synchronization, focal length re-adjustment, image registration, and color difference compensation. The discrepancies among smaller video tiles have to be resolved for seamless composition. This work demonstrates our effort towards this general goal. In this paper, we focus on the registration of two arbitrarily translated videos in the DCT domain under several simplifying assumptions. For example, temporal synchronization and focal length distortion problems have been well solved. We are not only concerned with the registration of the intra-coded frames (i.e. the I picture) but also the refinement which is based on the motion vectors extracted from P and B frames.

### III. PROPOSED ALGORITHM

#### A. System Overview

In this work, we assume that inputs to the system are two synchronized MPEG videos at a frame rate of 30 fps. Also, there are only translation differences between them, both containing some moving objects. In order to avoid the ambiguity caused by pure image-to-image alignment and trajectory-based alignment, the input sequences to the proposed system are first segmented into two parts: the static background and the moving objects. Then, they are separately registered based on their associated spatial and the temporal information.

The flow chart of the proposed algorithm is given in Fig. 1. The unit of the process is 1 GOP (15 fps in our experiments). In other words, the displacement parameters are updated for each GOP. Take the first GOP as an example. After applying four edge detectors to the DC map of the I frame, the first

set of alignment parameters is determined. Since this is the block-based alignment, a further refinement process is required in order to reach higher accuracy. In the second pass, the motion information of objects from each frame within the same GOP is used to obtain several other sets of refinement parameters. Based on alignment and refinement parameters, we can estimate the final displacement parameter using a weighted average of them.

#### B. Static Background Alignment

Based on the I frames of two input sequences, DC maps are available by extracting out the DC coefficients of all blocks in the luminance (Y) component. Since only the DC value is considered for each  $8 \times 8$  block, the size of the DC map is  $1/64$  of that of the original image. This means that the data we dealing with are much less than that in the traditional pixel-domain approach. Based on the information provided by those two DC maps, a rough alignment can be done by applying a DCT-domain registration algorithm as described below.

First of all, four different edge detectors ( $H_1$ ,  $H_2$ ,  $H_3$  and  $H_4$ ) are applied to each of them. They are:

$$H_1 = \begin{bmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{bmatrix} \quad H_2 = \begin{bmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{bmatrix}$$

$$H_3 = \begin{bmatrix} -1 & -1 & 2 \\ -1 & 2 & -1 \\ 2 & -1 & -1 \end{bmatrix} \quad H_4 = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}$$

They measure the variation of the image in vertical, horizontal, 45 degree, and 135 degree directions, respectively. Each detector can produce one difference map so that there are four difference maps of each input image. We use  $D_{ij}$  to denote the difference map of image  $i = 1, 2$  with edge detector  $H_j$ ,  $j = 1, 2, 3, 4$ . Difference maps are normalized so that all of their values fall between 0 and 1 for further processing. Note that  $H_1$ ,  $H_2$ ,  $H_3$  and  $H_4$  are the second-order derivative filters which can represent more obvious behavior if compared to the first-order ones.

For each pair of difference maps, a content adaptive threshold is determined to generate the corresponding binary maps, say  $B_{ij}$ ,  $i = 1, 2$  and  $j = 1, 2, 3, 4$ . The main purpose of this step is to filter out minor changes. It reduces confusion and speed up the following alignment step. Those remaining features help determine the alignment parameters more accurately and reduce the processing time since the unnecessary detailed information has been eliminated. Based on the four sets of obtained binary images, we determine the alignment parameter by computing two-dimensional normalized cross-correlation and the optimal parameter is determined at the position where the maximum value occurs in both vertical and horizontal directions. A coordinate conversion, scaled up by a factor of 8, is performed on those parameters due to the size difference between the original and binary images. Then, the final estimated alignment parameter,  $(a_i, a_j)$ , can be acquired by either averaging or simply choosing the best one from these four vectors.

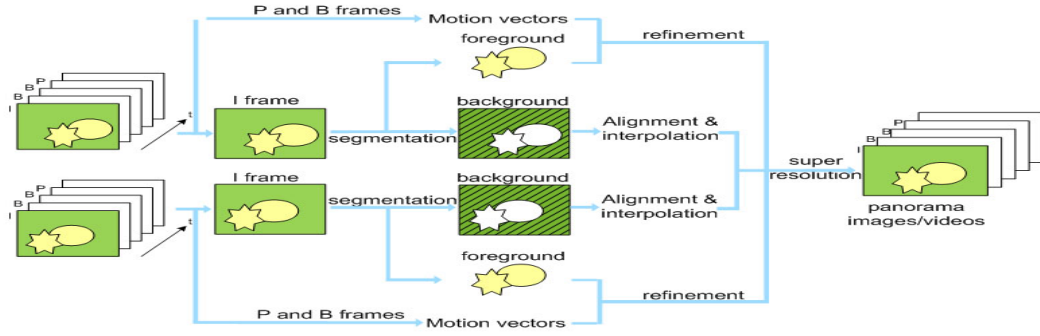


Fig. 1. The flow chart of the proposed system.

### C. Moving Object Alignment

In this step, the motion vectors of the major moving object obtained from all frames in one GOP are accumulated so that the trajectory can be formed. According to this information, a trajectory-based alignment process can be applied to enhance the alignment accuracy. Note that the diameter of the bouncing ball in our experiments is around 48 pixels, which corresponds to 3 macroblocks. Thus, in order to avoid incorrect information provided by motion estimation, the motion vector is not considered if its size exceeds a predetermined threshold value (which is set to 3 times the macroblock size in the given example).

Once the candidate motion vectors of each frame are determined, a one-dimensional correlation-based sequence alignment process is performed and the optimal parameter is determined at the position where the maximum value occurs. Since the GOP of the input sequence consists of 15 frames, we have 14 refinement parameters for each GOP in total, denoted by  $(r_{ik}, r_{jk})$ ,  $k = 1, 2, \dots, 14$ . There exists a tradeoff between the size of the moving object and the speed of the process. Usually, a larger moving object is preferred since it clearly and strongly represents the behavior of the cluster of macroblocks that contains the object. That is, it is easy to tell whether a macroblock belongs to the actual moving object or just an estimation error. However, in this case, we have to consider more motion vectors, which requires some more processing time. On the other hand, if the moving object is not that big, say within one macroblock, then only one motion vector is taken into consideration. Even though the computational complexity is lower, the robustness of the estimation result is also lower.

### D. Displacement Parameter Estimation

Following the procedures described in the last two subsections, coarse-alignment and motion-based refinement parameters,  $(a_i, a_j)$  and  $(r_{ik}, r_{jk})$ ,  $k = 1, 2, 3, \dots, 14$ , are obtained. The final displacement parameter,  $(d_i, d_j)$ , can be computed as

$$(d_i, d_j) = \alpha \times (a_i, a_j) + (1 - \alpha) \times \left[ \frac{1}{14} \times \sum_{k=1}^{14} (r_{ik}, r_{jk}) \right]. \quad (1)$$

In words,  $(d_i, d_j)$  is a weighted average of  $(a_i, a_j)$  and  $(r_{ik}, r_{jk})$ ,  $k = 1, 2, 3, \dots, 14$ . In our experiments, we tried different values of  $\alpha$  and found that  $\alpha = 0.5$  provides a reasonably good result.

The same procedure is applied to all GOPs of the input sequences. Note that the GOP of the generated input videos is 15 frames and since the frame rate is 30 fps (frames/sec), one can update the displacement parameters every I frame, *i.e.* every 0.5 sec. Thus, if there is an error occurring in P and B frames, it will not propagate for too long so that severe visual degradation of the output can be avoided.

If there is an abrupt scene change occurring in one GOP, the residual signal in one particular frame will become quite large. It is not difficult to find a threshold to detect such a scene change frame. Then, we are able to split the GOP into two separate parts. Thus, the proposed alignment process can be applied to each individual part separately.

## IV. SIMULATION RESULTS AND DISCUSSION

### A. Simulation Results

For the first example, the leading I frames of the two input MPEG2 sequences are shown in Fig. 2. As shown in this figure, the moving object is a yellow bouncing ball in front of a poster with a horizontal translation motion only. Fig. 3 shows the portion of the 15<sup>th</sup>, 30<sup>th</sup>, and 45<sup>th</sup> stitched frames from the two input sequences. The displacement parameters determined by the I frames and motion vectors of the first three GOP's are  $(410, 480)$ ,  $(408, 480)$ , and  $(410, 480)$ , respectively. Thus, we are able to use the background and motion information to do the alignment to generate a mosaic video of high quality.

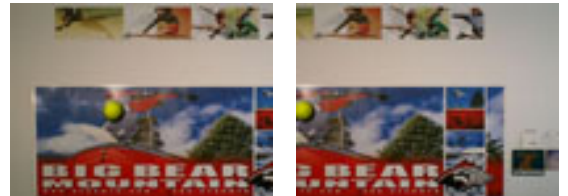


Fig. 2. The first frames of two input sequences for the 1st experiment.

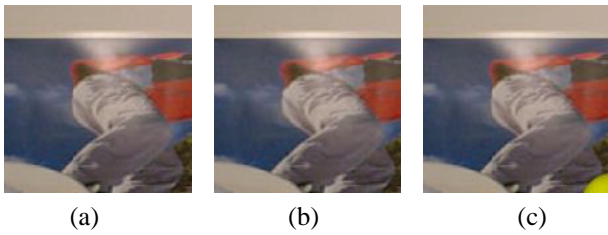


Fig. 3. The portion around the boundaries of stitched frames: (a) the 15th frame, (b) the 30th frame, and (c) the 45th frame.

The two input sequences for the second example are outdoor scene as shown in Fig. 4. The 15<sup>th</sup>, 30<sup>th</sup>, and 45<sup>th</sup> stitched frames are shown in Fig. 5. We see that these stitched frames have good quality. When comparing obtained displacements with the actual ones, we observe that the estimation errors are no larger than one half block (*i.e.* 4 pixels).

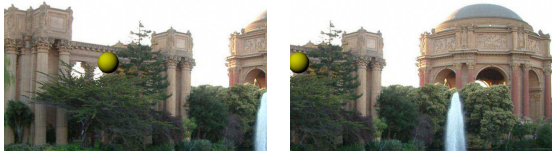


Fig. 4. The first frames of two input sequences for the 2nd experiment.



Fig. 5. The portion around the boundaries of stitched frames: (a) the 15th frame (b) the 30th frame and (c) the 45th frame.

### B. Discussion

Generally speaking, the proposed DCT-domain and motion vector-based alignment cannot provide sufficiently accurate information to reach 100% alignment accuracy since they are either block- or macroblock-based features. Some estimation error will result. However, by averaging and weighting, those effects can be reduced to some satisfactory degree. Also, our algorithm belongs to area-based alignment techniques, which is usually more robust than feature-point based alignment since some feature points may disappear in one of two frames and feature tracking is not easy.

Experimental results show that certain accuracy can be reached in the first step based on the alignment of DC coefficients alone in most cases. However, the proportion of the overlapping area to the original size and how many useful

features are within the overlapping parts would affect the quality of the composition. If there are few textured feature points in original images, the performance degrades. On the other hand, if the overlapping region contains highly regular periodic textured patterns, the accuracy of the alignment will decrease, too. In the second step, since only moving objects are considered, the characteristics of the objects plays an important role.

The two steps of the proposed algorithm are both conducted using coded video data. Thus, we do not have to seek additional image/video features and the tedious conversion between the spatial and compressed domain can be avoided. As a result, we can save a lot of computation. Also, since only DC coefficients are taken into consideration for rough alignment, it can be treated as a downsized version of the original image with the factor of 1/64. For those two reasons, the computation complexity is reduced a lot when it compared to the traditional spatial domain processes. This is the main advantage of the proposed algorithm.

### V. CONCLUSION

A DCT-domain registration technique for MPEG video was proposed and applied to the context of video mosaic. It was demonstrated by experimental results that the proposed method can reach certain degree of accuracy while the computational cost can be reduced dramatically as compared to the pixel-domain based techniques. The performance of the proposed method is consistent for both indoor and outdoor scenes. Although it is a block-based approach, the quality of the alignment has been enhanced to the half-block (4-pixel) accuracy. Some future extensions include the treatment of multiple moving objects with occlusion. The number of motion vectors required for good refinement is another interesting issue worth further investigation.

### ACKNOWLEDGMENT

The research has been funded in part by the Integrated Media Systems Center, a National Science Foundation Engineering Research Center, under the Cooperative Agreement No. EEC-9529152, and in part by KDDI Laboratories, Inc. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the National Science Foundation and KDDI Laboratories, Inc.

### REFERENCES

- [1] Lisa G. Brown, "A survey of image registration techniques," *ACM Computing Surveys*, (24)4: pp. 325–376, 1992.
- [2] J.-W. Hsieh, H.-Y. M. Liao, K.-C. Fan, M.-T. Ko, and Y.-P. Hung, "Image registration using a new edge-based approach," *Computer Vision and Image Understanding*, vol. 67, no. 2, pp. 112–130, Aug, 1997.
- [3] Aditi Majumder, Gopi Meenakshisundaram, W. Brent Seales and Henry Fuchs, "Immersive teleconferencing: a new algorithm to generate seamless panoramic video imagery," *Proceeding of the Seventh ACM International Conference on Multimedia*, October 30 – November 5, 1999.
- [4] Yaron Caspi, and Michal Irani, "A step toward sequence- to-sequence alignment," *CVPR*, 2000.
- [5] Barbara Zivota, and Jan Flusser, "Image registration methods: a survey," *Image and Vision Computing* 21, pp. 977–1000, 2003.