Proceedings of the 5th WSEAS International Conference on Telecommunications and Informatics, Istanbul, Turkey, May 27-29, 2006 (pp198-202)

Energy Based Video Synthesis

RANGA RODRIGO
The University of Western Ontario
Department of Electrical and
Computer Engineering
London, Ontario, N6A 5B9
CANADA

ZHENHE CHEN
The University of Western Ontario
Department of Electrical and
Computer Engineering
London, Ontario, N6A 5B9
CANADA

JAGATH SAMARABANDU
The University of Western Ontario
Department of Electrical and
Computer Engineering
London, Ontario, N6A 5B9
CANADA

Abstract: This paper describes an optimal method of inserting new frames or recovering missing frames in a video sequence. The method is based on an optimization scheme using graph-cuts that finds the 'optimal' frames to be inserted in between two given frames. The core problem is a typical visual correspondence problem between pixels in two or more frames and having formulated the appropriate energy, graph-cuts can be used for optimization. The two frames are assumed to be 'close' and the motion of the objects is small. The motion is seen as a set of two dimensional disparities, and the graph-cuts based optimization is able to find these. Once the disparities are found, an intermediate frame can be trivially placed at an arbitrary position in between the two original frames. The advantage of using graph-cuts instead of the typical techniques used in calculating optical flow lies in the global nature of the graph-cuts optimization. The success of our method is shown with synthetic and real image sequences. We show how the method can be extended to insert multiple frames in between the given two frames. One of the immediate applications is generation of synthetic slow-motion sequences.

Key–Words: Video, Frame Synthesis, Energy minimization

1 Introduction

In this paper, we address the problem of synthesizing new video frames based on existing neighboring frames. The challenge is to establish pixel correspondence between frames in the presence of moving or changing background and moving objects. In other words, given a pixel, we need to find the corresponding pixels in the given set of frames, in order to establish the path of the pixel within the spatio-temporal volume. Related but not equivalent problems are image completion, image and video texture synthesis [1], background removal and correction of corrupted video frames [2]. Although our focus in this paper is the frame synthesis problem, the methodology can be extended to handle those related problems as well.

The proposed method relies on the ability to solve the correspondence problem in a globally optimum manner. However, solving the correspondence is a difficult problem and we use the the graph-cuts based methods made available by Boykov *et al.* [3]. There are proven guarantees of convergence to a solution in the proximity of the global minimum depending on the energy function used. Once the correspondences are established across the given set of frames, the frame synthesis problem becomes a mere interpolation problem along the time axis to find the path of the pixel.

There are similar applications and methods of processing image sequences. Image sequence processing has become important due to the popularity of digital media. The applications include video noise removal, hole-filling and frame synthesis. Kokaram and Godsill [4] use a Markov chain Monte Carlo methodology to detect and remove additive noise components and thereby treat missing data in video. Global techniques have successfully been employed to solve the video completion problem. Such techniques tend to estimate the path of a pixel within the space time volume of the video sequence [2]. Once the paths of all the pixels which show motion are found it is possible to construct a new frame. Although the results of such methods are impressive, as the paths of a large number of points within the space time volume have to be calculated, they are computationally expensive. Local techniques of establishing pixel correspondence, although inferior to the global techniques in terms of the accuracy, have the distinct advantage of being fast. Our method being a global optimization scheme is capable of producing accurate results whilst being reasonably fast. This is a new application of using the ability of the graph-cuts optimization to globally estimate motion.

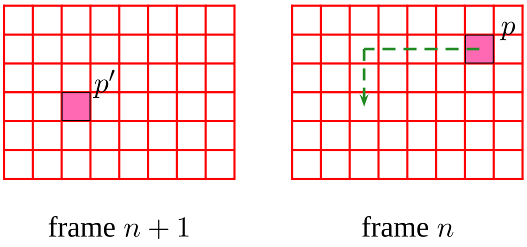


Figure 1: The pixel p in frame n is moved to pixel p' in frame $n+1$. For this pixel, motion parameters (disparities) are $f_p^h = 4$ and $f_p^v = 2$.

2 Method

The target video completion problem is casted as a visual correspondence problem. Corresponding pixels are found in two views (frames) taken at two instances of time. The two views are not similar due to motion. The 'new' frame is expected to preserve the integrity of the objects present in the previous and subsequent frames (see Figure 1).

2.1 Pixel Correspondence

The path of each pixel within the spatio-temporal volume should be found in order to synthesize a new frame. Figure 2 shows such an imaginary path. The main concern is that due to the dynamic nature of video making, simple cross–correlation matching techniques do not produce useful results. The reason is that such methods are inherently local. The proposed approach of solving this problem is to use the graph-cuts based labeling techniques to find the pixel correspondences in a globally optimum manner.

The energy function minimized using graph-cuts tends to assign a label to each pixel. The label assignment is done depending on the 'preference' of the pixel in question to be assigned with the label in question (called the data cost) as well as maintaining the piecewise smoothness of the label assignment (called the smoothness cost). There are two components in the energy function corresponding to these costs given by:

$$E(f) = E_{smooth}(f) + E_{data}(f), \qquad (1)$$

where f is the label and $E(f)$ is the cost associated with assigning label f. The corresponding energy function is of the form [3]

$$E(f) = \sum_{\{p,q\}\in N} V_{p,q}(f_p, f_q) + \sum_{p\in P} D_p(f_p). \qquad (2)$$

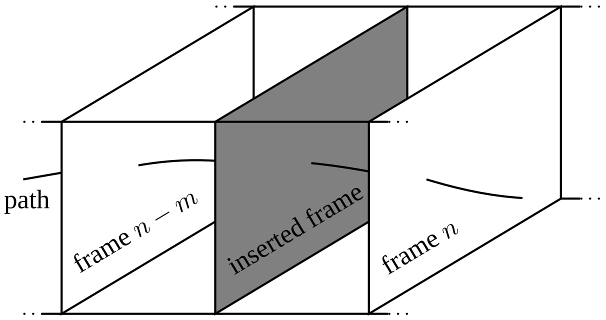


Figure 2: Path of a pixel within the spatio-temporal volume

Here the first summation represents the pairwise pixel interaction cost where N is the pixel neighborhood. $f_p \in L$ is the label assigned to pixel p. The second summation represents the data term where P is the set of possible pixels. In the context of motion estimation, the goal is to assign a label to each pixel denoting the motion exhibited by the point corresponding to the same pixel. Therefore, the possible labels are two dimensional disparity quantities representing the combination of vertical and horizontal motion. In other words, for a given pixel in one frame, the most similar pixel in the other frame is found within the limits of possible disparity labels. In addition to this pixel similarity, the labels are assigned in a smooth fashion due to the contribution of the first term in Equation 2. Interested readers are referred to the following references for details on how the energy minimization is taken place [5, 3, 6]. The energies are formed to incorporate the motion seen between two frames. The E_{data} part of the energy is calculated as a function of $(I_p - I'_{p+d})$ for every color plane, R,G and B. I'_{p+d} is the intensity of the pixel with a disparity d compared to pixel p in the first frame with intensity I_p. The E_{smooth} part is calculated as $e^{(f_p-f_q)^2/(2\sigma^2)}$. Here σ is the standard deviation of the image intensity. The energy quantities and σ are evaluated with respect to each of the color planes and combined appropriately. It is important to note that two dimensional disparities are required to be used and therefore f_p and f_q are two dimensional quantities. The algorithm that is capable of minimizing such energies is the so called swap algorithm [3]. Our current implementation permits to find pixel correspondences between a pair of images. Considering a pair at a time, correspondences across more than a pair can also be found.

2.2 Disparity Labels

The maximum amount of disparity between the two frames should be specified in advance for the graph-cuts based motion recovery algorithm. We assume a range of disparities from 0 pixels to the maximum disparity with increments of one. Maximum disparity should be chosen to include all the possible motions of pixels. However, the least should be selected as the computation time increases with the number of labels. In our experiments, these maximums are less than 10 pixels for both the synthetic and real images.

2.3 Frame Synthesis

New frames are synthesized depending on the pixel correspondences made as described in Section 2.1. Assuming that pixel concerned in frame n is p_n and the pixels correspondences have been made across m frames, pixels $p_{n-m}, p_{n-m+1}, \ldots, p_n$ correspond to each other. Since the coordinates x_n, y_n of each of these are available, it is possible to fit a curve and find the path of pixel p within the spatio-temporal volume. In the current implementation, we consider only a pair of neighboring pixels and therefore $m = 1$. This simplifies the calculation and we consider linear interpolation of pixel position. With these assumptions the value of the pixel p on the newly synthesized frame is:

$$I^{new}_{p,\alpha} = \begin{cases} I_p & \text{if } d_p = 0 \\ I_{p+d/\alpha} & \text{if } d_p \neq 0. \end{cases} \qquad (3)$$

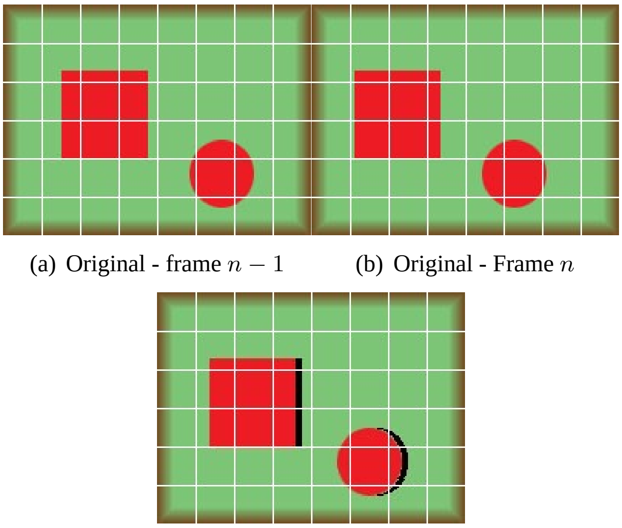
where d_p is disparity (label) assigned to pixel p and $\alpha \in (0, 1)$ controls where the new frame is placed. By selecting equispaced values for α more than one new frame can be generated.

3 Results

The method described in Section 2 is used for frame insertion for both synthetic and real sequences. Following assumptions are made, without loosing generality, to make the motion calculation step simpler and faster:

1. No significant motion exists in vertical direction.
2. Background is still and therefore can be used for filling in occluded portions (camera is still).
3. The moving objects move only in left direction.

Figure 3 shows the results obtained using a synthetic pair of frames. The square and the circle are the objects that have moved. The graph-cuts based correspondence algorithm is able to accurately assign labels to those pixels which have moved. The new frame is inserted midway between the two original frames using Equation 3. Figure 4 shows the results of inserting three frames. The motion of the person is accurately reconstructed, particularly observable from the motion of features such as legs and arms. This shows the potential of our algorithm in the area of generating slow-motion sequences.


(a) Original - frame $n-1$ (b) Original - Frame n

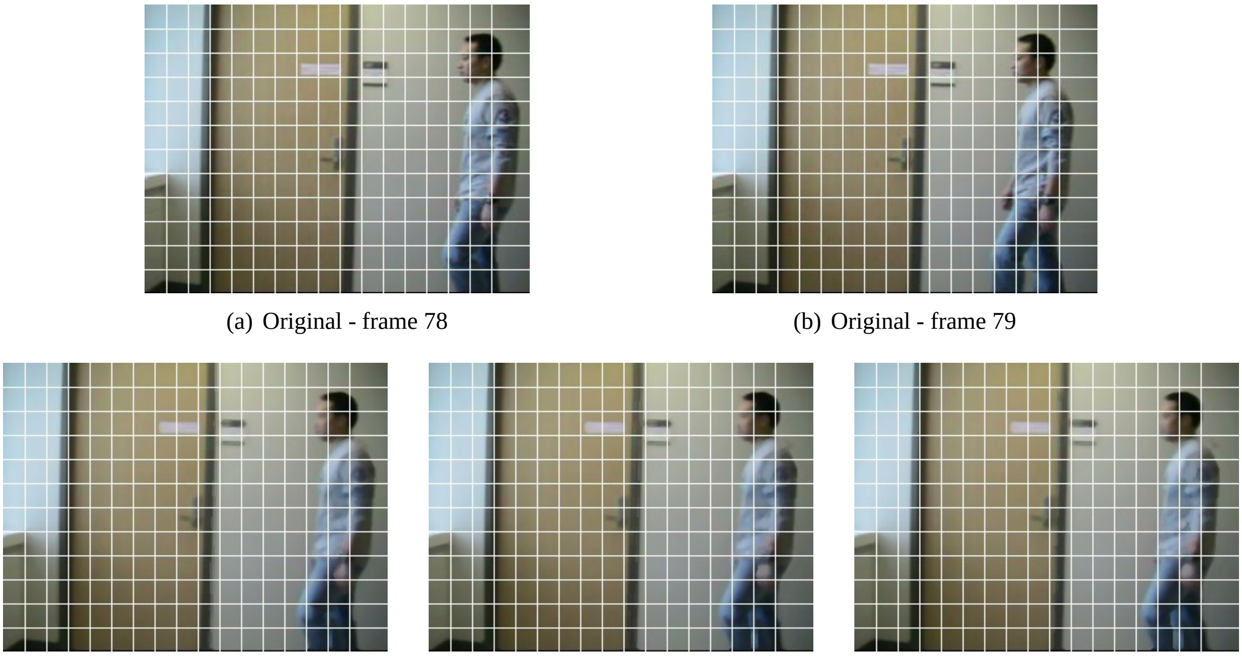
(c) Newly inserted frame

Figure 3: Results with synthetic images (a) and (b) are two adjacent frames and (c) is the new inserted frame. The dark regions to the right of the objects are retained to accentuate motion representation. A grid is manually placed for easy comparison.

The method is applied to real video frames as shown in Figure 5. The two original frames, the newly generated frame and the motion image are shown. The two original frames are 9 frames apart in a typical 30 frames per second sequence obtained using a commercial camcorder at a resolution of 320×240. The motion has been recovered up to a satisfactory degree of accuracy. The scene itself is a comparatively difficult scene for the algorithm, with the moving person occupying a large portion of the image. Moreover, the motion is non-rigid, specially the motion of the clothes worn by the person. Although the newly generated frame is therefore meaningful here, when the original frames are far apart, the motion recovery may be inaccurate.

4 Conclusion

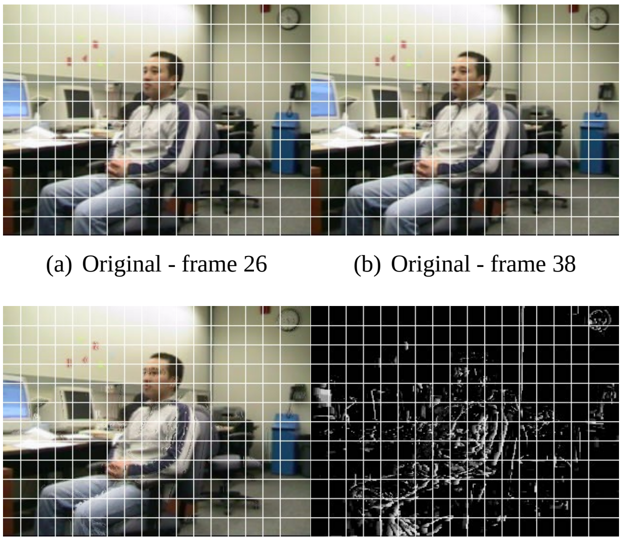
In this paper we have presented a system with the ability of using an energy minimization scheme to synthesize a new video frame within a given set of frames. Our method relies on the ability of energy minimization based on graph-cuts to assign disparity labels to pixels representing motion. The pixel path within the video volume is found using the pixel correspondences. New frames are synthesized by an appropri-



(a) Original - frame 78 (b) Original - frame 79

(c) New frame 1 (d) New frame 2 (e) New frame 3

Figure 4: Synthesis of 3 intermediate frames with real images. (a) and (e) are two adjacent frames and (b), (c) and (d) are the newly inserted frames after light median filtering. A grid is manually placed for easy comparison.



(a) Original - frame 26 (b) Original - frame 38

(c) Inserted (d) Motion

Figure 5: Results with real, comparatively difficult images (a) and (b) are two adjacent frames and (c) is the new inserted frame. (d) shows the motion between (a) and (b). Higher the intensity larger is the motion. A grid is manually placed for easy comparison.

ate interpolation. Synthetic and real results show the success of the method. This approach is a novel application of graph-cuts based optimization to successfully solve the frame synthesis problem. However, the processing time for a typical frame is in the range of seconds and therefore the method is suitable only for off-line processing.

4.1 Future Work

The current implementation has been done with a set of simplifying assumptions stated in Section 3. We plan to produce results with less restrictive assumptions and will be made available on our web page http://mahaweli.eng.uwo.ca/iris/vidcompl.html. There are a couple of areas where further investigation is required. First, methods of obtaining the pixel path in a straight forward manner, instead of matching pairs of frames at a time will be interesting. Second, the possibility of generalizing our system to handle situations such as object removal from video and restoring corrupted video should be investigated.

References:

[1] V. Kwatra, A. Schödl, I. Essa, G. Turk, and A. Bobick, “Graphcut textures: Image and video synthesis using graph cuts,” in *Proceedings of ACM*

SIGGRAPH 2003, San Diego, CA, July 2003, pp. 277–286.

[2] Y. Wexler, E. Shechtman, and M. Irani, "Space-time video completion," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Washington, DC, June–July 2004, pp. 120–127.

[3] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222–1239, November 2001.

[4] A. C. Kokaram and S. J. Godsill, "Mcmc for joint noise reduction and missing data treatment in degraded video," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 189–205, February 2002.

[5] O. Veksler, "Efficient graph-based energy minimization methods in computer vision," Ph.D. dissertation, Computer Science, Cornell University, 1999.

[6] Y. Boykov and V. Kolmogorov, "An experimental comparison of min–cut/max–flow algorithms for energy minimization in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1124–1137, September 2004.