The final published version is available at http://dx.doi.org/10.1109/ICME.2002.1035670.

REDUCING BLURRING-EFFECT IN HIGH RESOLUTION MOSAIC GENERATION

Ramazan Savaş Aygün and Aidong Zhang

Department of Computer Science and Engineering
State University of New York at Buffalo
Buffalo, NY 14260-2000
{aygun,azhang}@cse.buffalo.edu

ABSTRACT

The mosaic generation methods benefit from recent global motion estimation (GME) methods, which yield almost accurate estimation of motion parameters. However, the generated mosaics are usually more blurred than original frames due to image warping stage and errors in motion estimation. The transformed coordinates resulting from GME are generally real numbers whereas images are sampled into integer values. Although GME methods generate proper motion parameters, a slight error in motion estimation may propagate to subsequent mosaic generation steps. In this paper, we propose a method to generate clearer mosaics from video. The temporal integration of images is performed using the *histemporal* filter based on the histogram of values within an interval. The initial frame in the video sequence is registered at a higher resolution to generate high resolution mosaic. Instead of warping of each frame, the frames are warped into the mosaic at intervals. This reduces the blurring in the mosaic.

1. INTRODUCTION

Mosaic generation has been studied for both content-based retrieval and video compression. MPEG-4 [1] enables decoding and encoding of layered sprites for the objects and the background. The different types of mosaics and mosaic generation methods are covered in [2, 3]. The initial stage of a mosaic generation is Global Motion Estimation (GME). The motion is usually modeled using perspective, affine, translation-zoom-rotation, or translational motion models. Most of the GME techniques concentrate on the accuracy of motion parameters of the chosen motion models [4, 5, 6]. These methods usually include an initial estimation of the subset of the motion parameters and then adjusting of the motion parameters using a hierarchical pyramid of low-pass filtered images.

Different representations of mosaics like static, dynamic and synopsis mosaic have been investigated in [2]. Direct methods are applied to align images and to generate the mosaic. A sprite creation method based on connected operators is presented in [7]. A detailed work on estimation of motion parameters and generation of sprites has been presented in [5]. A high resolution mosaic is generated by sliding the mosaic and warping the next frame into the mosaic [6]. Since warping occurs for every frame, the generated mosaic can still be blurred. Moreover, temporal integration methods are used according to the type of the mosaic that will be generated. The temporal integration methods also causes blurring in the mosaic.

In this paper, we propose a method for generating high resolution mosaic from video. The frames are integrated using the *histemporal* filter. The *histemporal* filter is a generalized filter and keeps the histogram of values that map to a specific interval. The initial mosaic is maintained at a higher resolution to reduce the blurring due to real-valued transformed coordinates. The frames are warped into the mosaic at intervals. Since warping of frames is performed using bilinear interpolation, a low-pass effect is introduced. Therefore, ignoring unnecessary frames yields clear mosaic generation. After developing our method, experiments are conducted on standard MPEG test sequences.

This paper is organized as follows. The motion estimation is explained in Section 2. High resolution mosaic generation and *histemporal* filter are presented in Section 3. The experiments and results are reported in Section 4. The last section concludes our paper.

2. MOTION ESTIMATION

The mosaic should include every section that is visible throughout the video sequence. If there is no a priori motion information for a video sequence, the motion has to be estimated between each sequential frame.

There are different types of motion models that are used in GME depending on the camera operations and the structure of the scene. In this paper, we detect camera motion

This research is supported by NSF grant IIS-9733730 and NSF grant IIS-9905603.

that is parameterized by perspective motion model:

$$x'_i = \frac{a_0 + a_2 x_i + a_3 y_i}{a_6 x_i + a_7 y_i + 1} \quad y'_i = \frac{a_1 + a_4 x_i + a_5 y_i}{a_6 x_i + a_7 y_i + 1} \tag{1}$$

where a_0, a_1, a_2, a_3, a_4, a_5, a_6, and a_7 are motion parameters and (x'_i, y'_i) is the transformed coordinate for (x_i, y_i). This model turns into affine motion when $(a_6 = 0,\ a_7 = 0)$, translation-zoom-rotation motion when $(a_4 = -a_3,\ a_5 = a_2,\ a_6 = 0,\ a_7 = 0)$, and translational motion when $(a_2 = 1,\ a_5 = 1,\ a_4 = 0,\ a_5 = 0,\ a_6 = 0,\ a_7 = 0)$.

The error between two frames can be declared as

$$\varepsilon = \sum^N e_i^2 \tag{2}$$

where $e_i = I'(x'_i, y'_i) - I(x_i, y_i)$, $I(x_i, y_i)$ is the intensity at (x_i, y_i) in the previous frame, and $I'(x'_i, y'_i)$ is the intensity at the transformed coordinate in the current frame. Error ε is computed for pixels overlapping in two frames.

The iterative descent methods are likely to be trapped in local minima. Our mosaic generation method skips some of the frames to reduce blurring in the mosaic. The motion estimation is performed between each sequential frame and also between frames at specific intervals. Motion estimation between farther frames is more prone to errors due to the initial estimation and possible large displacement. Accumulated motion parameters or relative motion with respect to the initial frame in the interval is a good approximation of the motion parameters. In matrix form, affine motion estimation can be written as

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_2 & a_3 \\ a_4 & a_5 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} a_0 \\ a_1 \end{bmatrix}. \tag{3}$$

More generally, this can be written as

$$v' = Mv + t \tag{4}$$

where M contains the motion parameters for the first matrix and t contains the translational parameters. The relative motion is computed as

$$v'' = M'(Mv + t) + t' = M'Mv + (M't + t') \tag{5}$$

where v'' is the vector for the new transformed coordinates; M' and t' hold the current motion parameters; and M and t hold the motion parameters up to the current frame.

To increase the robustness of motion estimation, we use M-estimators [4, 5] and the error is expressed as:

$$\sum^N \rho(e_i) \tag{6}$$

where $\rho(e_i) = e_i^2$ in the original formulation. Since this function gives more weight to large errors, it is biased by local motion (which are outliers for global motion). To decrease the effect of outliers, the truncated quadratic motion is used:

$$\rho(e_i) = \begin{cases} e_i^2, & \text{if } |e_i| \le t \\ 0, & \text{if } |e_i| > t \end{cases} \tag{7}$$

where t is a threshold selected according to the histogram of the errors.

3. MOSAIC GENERATION

3.1. Histemporal Filter

The linear temporal filters like averaging, recursive filters like Kalman filter [8], and order-statistic filters like median filters have been used for noise reduction or removal in image sequences. Temporal averaging yields a blurred mosaic, if the video includes moving objects or motion cannot be estimated accurately. Median filters require enormous storage to detect the temporal median filter. Moreover, temporal median may yield erroneous results, if the expected median can take several values. For example, the frequency of pixel values 100 and 101 is 20 and 22 for a pixel coordinate in the mosaic, respectively. Although this difference may result from the illuminance change in the environment, they are treated as different. If the frequency of another pixel value is 25, this value will be chosen by mistake. In fact, averaging (of 100 and 101) would yield a better result. *Histemporal* filter is a *temporal* filter that is based on the *histogram* of intensity values within a specific interval.

The $interval$ determines the precision of temporal integration in mosaic generation. For a 8-bit per pixel grayscale image, all the pixels lay in $[0, 255]$. There will be $\lceil \frac{256}{interval} \rceil$ slots in the histogram. If $interval$ is 256, histemporal filter becomes temporal averaging. If $interval$ is 1, the temporal interval becomes temporal median filter.

Two data structures are used to obtain the histemporal filter: *frequency* array and *average* array. *Frequency* array keeps the frequency of each interval of the histogram. As the frames are processed, the frequency of an interval is increased for each pixel value belonging to the interval. *Average* array maintains the average of the values as new values are processed for each slot. Histemporal filter returns the average value of the interval having the highest frequency. Figure 1 (a) shows a histogram where $interval$ is 16. The interval $[81, 96]$ has the highest frequency. Figure 1 (b) displays the frequencies of the values that lay in this interval. During histemporal filter computation, the average of these values is computed as they arrive.

3.2. High Resolution Mosaic Generation

Motion parameters that are obtained from Equation 1 are usually real numbers and yield real-valued transformed coordinates. The original images are sampled into integer do-

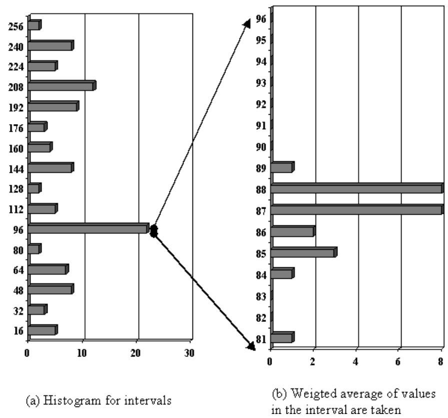

Figure 1: Histemporal filter.

main. The ordinary techniques create a mosaic having a resolution of the initial frame in the sequence. The pixel locations in the mosaic may not correspond to the integer-valued pixel locations in the new frame. Approaches like bilinear interpolation are used to estimate the pixel value at the location. Bilinear interpolation takes the weighted average of the closest pixels and blurs the image.

A high resolution video mosaicking approach is proposed in [9]. A high resolution mosaic is generated where a mosaic also contains half-pel data. When a new frame is processed, a shift (diagonal, vertical, or horizontal) on the mosaic is assumed, and the frame is warped into the corresponding area in the mosaic. This usually preserves the original sharpness of the image. However, this approach does not consider the precision of the transformed coordinates and warping still occurs at a low resolution because of shifting. In our case, warping occurs at high resolution (Figure 2). Every pixel in the warping region is updated during warping.

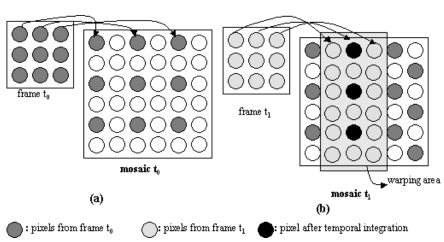

Figure 2: High resolution mosaic.

The motion parameters are also affected by the moving objects and aperture problem. This causes some deviation from the original values of motion parameters. When the motion estimation is performed from frame to frame, the error accumulates and propagates to the later motion estimation and warping. In addition, warping at every frame also introduces blurring. Thus, instead of warping at every frame, the frames are warped into the mosaic at intervals. But the motion estimation has to be performed for each sequential frame. The previous frame is mapped from the mosaic to avoid error accumulation. Three thresholds are used: maximum accumulated displacement (mad), maximum scale factor (msf) and maximum interval length (mil). The motion between consecutive frames are accumulated until the displacement is less than mad and scale (zooming) factor is less than msf. Otherwise, the motion between the first frame and the last frame in the interval may increase significantly, and motion estimation methods may yield less accurate parameters. If there is no significant motion in the sequence, the relative motion is computed for at most mil frames. This upper-bound is needed to remove the objects from the background mosaic. The frame is also warped into the mosaic when the direction of camera motion changes.

4. EXPERIMENTS

In our experiments, the resolution of the mosaic is twice as the initial frame of the sequence, thus resulting in half-pel accuracy. The mad and mil are both selected as 10. Figure 3 shows an ordinary blurred mosaic generated from 'coastguard' MPEG test sequence. If the motion can be modeled using translational model, the images can be warped according to the precision of transformed coordinates. MPEG-4 test sequence 'coastguard' can be modeled using translational model. Figure 4 shows the high resolution background mosaic generated after 300 frames. No segmentation mask is used in the generation. The water texture is smoothed because of temporal texture, and has been removed from the mosaic. The right side of the figure includes parts that are not filled by frames. Therefore the right side looks darker. These locations are filled with bilinear interpolation. The smoothed regions in the ordinary mosaic are clearly visible in the high resolution mosaic.

Figure 3: Ordinary Mosaic.

There is no standardized performance tests for generating mosaics. The most common method is averaging PSNR values for a video. Although PSNR is a good indication of similarity between images, average of PSNR values is not always a good measure for video. Figure 5 shows the mosaic generated for 'foreman' MPEG test sequence from frames 195 to 240. The corresponding PSNR values for frames that are generated from high resolution mosaic and ordinary mosaic are given in Figure 6. We have used affine motion model for 'foreman' sequence.

Figure 4: High resolution mosaic from coastguard.

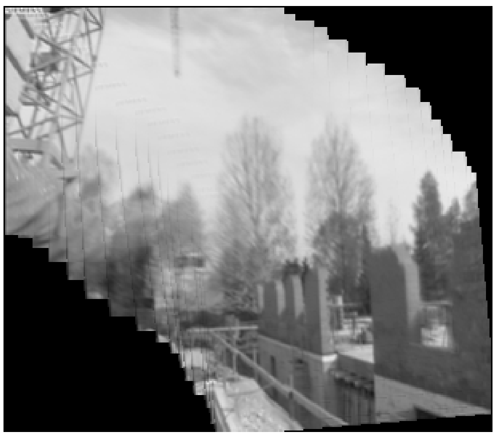

Figure 5: Mosaic generated from 'foreman'.

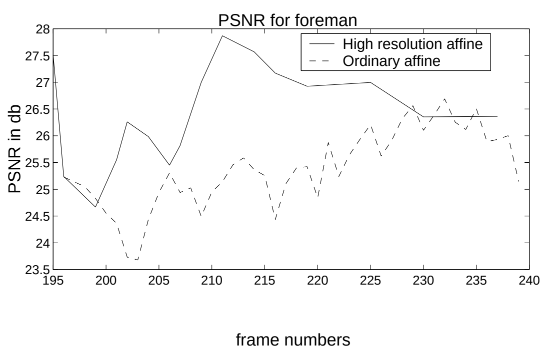

Figure 6: PSNR for foreman sequence.

5. CONCLUSION

In this paper, we presented a method for high resolution mosaic generation from video. Motion estimation is performed between each consecutive frame not to miss visible areas in the sequence for mosaic generation. The blurring in the mosaic generation is reduced by warping at intervals and at a higher resolution. Although high resolution mosaic warping increases elapsed time, this is compensated by warping at intervals. The number of frames that are skipped can further be decreased, but this is left as a further research.

6. REFERENCES

[1] T. Sikora, "The mpeg-4 video standard verification model," *IEEE Trans. Circuits Syst. Video Technology*, vol. 7, pp. 19–31, February 1997.

[2] M. Irani and P. Anandan, "Video indexing based on mosaic representations," in *Proceedings of IEEE*, May 1998, pp. 905–921.

[3] R. Szeliski and H-Y. Shum, "Creating full view panoramic image mosaics and environment maps," in *Computer Graphics Proceedings, Annual Conference Series*, 1997, pp. 251–258.

[4] F. Dufaux and J. Konrad, "Efficient, robust, and fast global motion estimation for video coding," *IEEE Transactions on Image Processing*, vol. 9, no. 3, pp. 497–501, March 2000.

[5] T. Sikora A. Smolic and J.-R. Ohm, "Long-term global motion estimation and its application for sprite coding, content description and segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 8, pp. 1227–1242, December 1999.

[6] A. Smolic and J.-R. Ohm, "Robust global motion estimation using a simplified m-estimator approach," in *Proc. ICIP2000, IEEE International Conference on Image Processing*, September 2000.

[7] P. Salembier, O. Pujol, and L.Garrido, "Connected operators for sprite creation and layered representation of image sequences," in *IV European Signal Processing Conference*, September 1998, pp. 2105–2108.

[8] O. Munkelt C. Ridder and H. Kirchner, "Adaptive background estimation and foreground detection using kalman-filtering," in *Proceedings of Interenational Conference on Recent Advances in Mechatronics*, June 1995, pp. 193–199.

[9] A. Smolic and T. Wiegand, "High-resolution video mosaicing," in *Proc. ICIP2001, IEEE International Conference on Image Processing*, October 2001.