EVALUATION OF IMAGE STABILIZATION ALGORITHMS

Carlos Morimoto

IBM Almaden Research Center
650 Harry Rd
San Jose, CA 95120

Rama Chellappa

Department of Electrical Engineering
University of Maryland
College Park, MD 20742

ABSTRACT

Several techniques for electronic image stabilization have recently been proposed, but very little research has been done to compare and evaluate such techniques. In this paper we propose a set of measures to evaluate image stabilization algorithms based of their fidelity, displacement range, and performance. These measures do not require calibration or ground truth, making the evaluation procedure very simple and flexible, i.e., it provides the means to compare techniques based on different motion models. We have used this procedure to compare several image stabilization algorithms and also evaluate the sensitivity of these algorithms to some of its parameters. These same procedures could also be used for the comparison and evaluation of motion estimation and image registration techniques.

1. INTRODUCTION

Electronic image stabilization (EIS) is the process of generating a compensated video sequence where any and all unwanted camera motion is subtracted from the original input. Most proposed EIS systems compensate for all motion [3, 4, 6, 7, 8, 10], producing a sequence where the background remains motionless. Other techniques only subtract the 3D rotation of the camera [5, 9, 11], generating a derotated sequence.

Since motion estimation is the main component of an EIS system, the evaluation of the system could be based on the performance of the motion estimation module alone, in which case one could use synthetic or calibrated sequences where the inter-frame motions are known [2].

Aside from the issue of generating sequences with known motion, most EIS systems use approximate parametric global transformations, which creates the problem of finding the optimal transformation from the ground truth data, so that the motion estimates can be evaluated in terms of a distance measure from these optimal parameters. Another important issue is how to compare the performance of systems based on different motion models, since distance measures might be model-dependent.

Other methods of evaluating image stabilization systems are presented in [1, 8]. [1] compares the performance of different stabilization algorithms based on the accuracy of a real-time object tracker, and [8] considers the maximum displacement velocity in pixels per second, computed as the product of the frame rate and the maximum image displacement between frames.

The support of the Defense Advanced Research Projects Agency (ARPA Order No. A422) and the U.S. Army Research Office under Grant DAAH04-93-G-0419 is gratefully acknowledged

In this paper, we evaluate the fidelity of image stabilization techniques using the peek signal-to-noise ratio (PSNR) between stabilized frames. This method does not require the use of calibrated sequences to compare different systems, and provides a simple means of comparing systems based on different motion models. Synthetic sequences are used to measure other system properties, such as the range of displacements within which they operate.

2. IMAGE STABILIZATION ALGORITHM

A general image stabilization algorithm is composed of a motion estimation module (ME), a motion compensation module (MC), and an image composition module (IC), as shown in Figure 1. ME estimates the motion between frames, and sends the motion parameters to MC, which computes the global transformation necessary to stabilize the current frame. IC then warps the current frame according to that transformation, generating the stabilized sequence, and possibly image mosaics.

Figure 1: Modules of a general EIS system.

The EIS algorithms evaluated in Section 4 are based on the system described in [8], which uses a multi-resolution feature-based motion estimation technique that fits a four parameter similarity model to the feature displacements, and then combines the interframe motion to generate the transformation which stabilizes the current frame. We have extended the algorithm to include the Euclidian and affine motion models.

The three parameter Euclidean model compensates for translations and rotation around the optical axis. The similarity model has an extra parameter to include scaling; and the affine model requires 6 parameters to accommodate different horizontal and vertical scaling, and skewing. In Section 4 we compare the performance of these models for several real and synthetic image sequences, and also the behavior of the algorithm when some of its parameters are changed.

3. EVALUATION PROCEDURES

We have designed three evaluation procedures to measure the fidelity, displacement range, and performance of EIS algorithms. *Fidelity* is a measure of how well stabilization is compensating the motion of the camera, i.e., how precisely the motion model fits

0-7803-4428-6/98 $10.00 © 1998 IEEE

the actual camera motion. *Displacement range* is defined by the minimum and maximum image displacements supported by the stabilization system; and *performance* is defined by the maximum displacement velocity which the system can compensate for, in pixels per second, given by the product of the frame rate and the maximum interframe translational displacement.

3.1. Fidelity

Intuitively, when all motion is compensated for, there should be no residual motion after stabilization, i.e., the difference between two stabilized images should be zero for every pixel. Divergence from zero can be caused by noise, estimation errors, distortions due to limitations of the motion model and interpolation during warping, etc. For stabilization purposes, the PSNR can be considered as a measure of the departure from the optimal case, or as a measure of the overlap between two images, which is maximized when the images are identical.

The PSNR between stabilized frames is used to measure the fidelity of a system. The PSNR between I_1 and I_0 is defined as

$$PSNR(I_1, I_0) = 10 \log \frac{255^2}{MSE(I_1, I_0)} \quad (1)$$

where the mean squared error (MSE) is a measure of the average departure per pixel from the desired stabilized result. The PSNR gives a relation between the desired output and the residual image, in terms of their powers (for gray images with a maximum intensity of 255). The higher the PSNR between two stabilized frames, the better the fidelity of the system.

The above formulation does not account for non-overlapping regions where compensation cannot be done. If the PSNR were computed just over the overlapping regions, the fidelity measure would not be meaningful for small overlapping areas. In order to handle these cases, nonoverlapping pixels are copied from the current frame before computing the PSNR. The worst scenario occurs when the global transformation warps the image completely off the reference frame, and for this case a lower bound (LB) can be defined as the PSNR between the reference and current frames without compensation. We assume that the system has produced a valid output whenever the PSNR between stabilized frames is higher than LB (or sometimes LB plus a constant offset). Erroneous motion estimates can in fact produce PSNRs below LB.

To implement this procedure, some routines of the IC module have to be modified to account for the non-overlapping areas. Given a particular stabilization system and an arbitrary sequence, two measures are computed. The first is a measure of the interframe transformation fidelity (ITF), and the second measures the global transformation fidelity (GTF). ITF is defined as the PSNR between two consecutive stabilized frames, and GTF corresponds to the PSNR between the reference frame and the current stabilized frame. The lower bounds on ITF and GTF will be respectively denoted by LB_i and LB_g.

3.2. Displacement Range

The second procedure determines the range of displacements supported by a system. Some synthetic image sequences were created for this procedure as follows: given a single image I of large dimensions, a window of smaller size (e.g. 128×128) is first placed at a fixed position on the image. This window is used to compose the output sequence. The first frame is defined by the window itself, and the displacement velocity increment (acceleration) is set to zero. The following frames are created by incrementing the displacement velocity by a certain amount (acceleration), and warping I according to the new displacement velocity using bilinear interpolation. As a result, the contents of the window, when placed on the warped image, change proportionally. The precision of this measurement is defined by the acceleration step between frames.

For very small displacements, the PSNR between consecutive frames is very high, i.e., LB_i is very high. If the errors in the estimated parameters are bigger than the true transformation parameters, ITF is lower than LB_i. As the displacement increases, LB_i decreases and ITF increases if the displacement is large enough to be estimated. Therefore, one curve eventually crosses the other. This crossing point is used to define the minimum image displacement for which the system can compensate.

To determine the upper bound on the range of displacements, some synthetic image sequences were created using larger acceleration steps between frames. It is expected that when the system is working properly, ITF remains higher than LB_i, which is low for large inter-frame displacements. When the displacement is too large, the system produces invalid motion estimates, causing ITF to drop and possibly become lower than LB_i. We define the maximum image displacement to be the point where the ITF curve crosses the LB_i curve.

3.3. The Performance Measure

Frame rate is also an important feature of EIS systems, but this measure alone might be misleading since the robustness and accuracy of the system can be easily sacrificed to increase speed. Performance will be defined as the maximum displacement velocity supported by the system, which is defined by the product of the frame rate and the maximum translational displacement, in pixels per second.

4. EXPERIMENTAL RESULTS

We used the algorithms mentioned in Section 2 for all experiments. These standard algorithms were configured to use the same settings for all parameters, such as the number of feature points, pyramid levels, search window sizes, etc. Sixteen features were automatically detected and tracked using search windows of size 5 pixels per pyramid level. Two pyramid levels were constructed for images of resolution 128×128, and three levels were used for images of higher resolution. All synthetic sequences were of resolution 128×128, and all real uncalibrated sequences were of resolution 320×240.

Figure 2 shows the results of evaluating the three systems using two uncalibrated sequences. The first column shows the results for the UGV sequence, which is composed of 30 frames of real video, where the camera starts zooming out and than pans from right to left. The graph on top of the first column shows the ITF and LB_i for all frames. Observe that the results for the affine- and similarity-based systems are very similar, while the Euclidean system performs poorly during the first ten frames, which correspond to the zooming part of the sequence. This result is expected since the Euclidean group does not compensate for scaling.

After the 20th frame, the sequence does not overlap the reference frame. This can be observed from the GTF curves shown in the bottom graph in the first column of Figure 2. The GTF drops from frame to frame since each new frame has less overlap with the

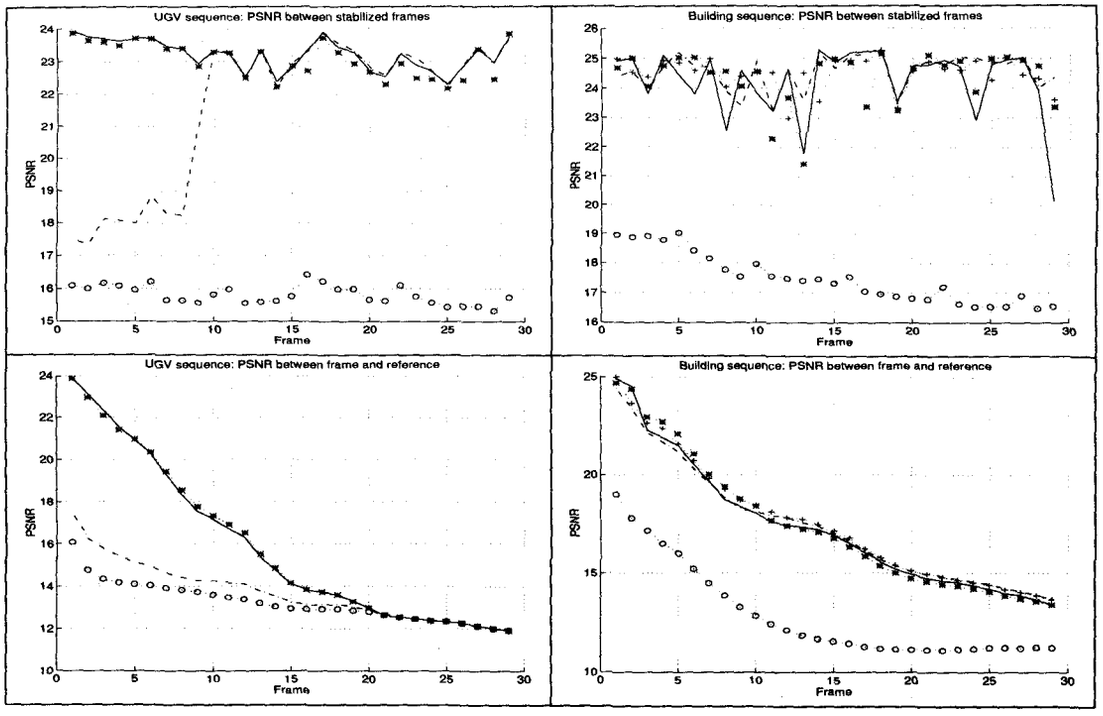

Figure 2: Results from Experiment 1. The (.*.) curve shows the results for the affine fit, (.+.) for affine using 20 feature points, (-) for similarity, (-.) for Euclidean, and (.o.) for lower bound.

reference frame. The GTF of the Euclidean system is considerably smaller due to the lack of scaling compensation.

The second column of Figure 2 shows the ITF and GTF for the Building sequence. This sequence is also composed of 30 frames of real video, and contains a simple pan from left to right. In this case, since there is no change in scale, all the curves are very similar, i.e., all systems perform equally well. It is important to notice from the ITF graph that feature outliers have much more influence on the performance of higher-order models. To test this hypothesis, the affine system was reconfigured to use 20 features instead of 16 (shown by the (.+.) curve); the performance improvement can be seen from the ITF and GTF curves. Since the same 16 features are used for all systems, the least-square fit seems to be more robust for the lower-complexity models. Both the UGV and Building sequences are available in MPEG format at http://www.cfar.umd.edu/~carlos/ICASSP98/evaluation.html.

Figure 3 shows the results of determining the minimum displacement for each system. Two synthetic sequences composed of 19 frames each were created for this experiment. The inter-frame acceleration step was set to 1/10th of a pixel/frame^2, i.e., frame F_n has a displacement of $n/10$ pixel from F_{n-1}, for $n > 0$. The original (Bahia and Boat) sequences are available at the same URL address. For this experiment we introduced a fourth system based on the Euclidean group, but with a simpler grid-to-grid (no subpixel precision) feature tracker. The measurements for this system are shown by dotted lines with crosses (.+.). For the standard systems, ITF becomes larger than BL_i after the second frame, i.e., the minimum displacement is below 0.3 pixel/frame. For the new system, the minimum displacement is below 0.6 pixel/frame for the Boat sequence, and below 0.7 pixel/frame for the Bahia sequence.

Figure 4 shows the results of determining the maximum displacements. Similar sequences were created, now with an acceleration of 1 pixel/frame^2. The maximum displacement is a property of the system related to the search sizes and number of pyramid levels. The search sizes were set to ± 5 pixels on each level, and two pyramid levels were constructed. A fourth system with the same search window size but only one pyramid level (without multi-resolution) was also tested; its results are shown by the (.+.) curves.

For the Boat sequence, the maximum displacement of the standard systems lies between the 13th and 14th frame, so that it is safe to say that it is above 13 pixels. For the Bahia sequence, which is of lower quality, this also seems to be the case, although there is a considerable drop after the 10th frame. For the system using only one pyramid level, the maximum displacement lies around 5 pixels, which corresponds to the size of the search window. The analysis of minimum and maximum displacements is not limited to translations. It is simple to create a synthetic sequence by varying any parameter of a transformation group, and then empirically determining the system's operating range for the sequence.

All systems were implemented on a real-time image processing platform, and they are able to process images of resolution 128×120 pixels at 17 frames per second (for the above standard configuration), i.e., these systems are able to stabilized camera motions of up to 221 pixels/second.

5. CONCLUSION

We have proposed a simple procedure for evaluating the fidelity, range of displacements, and performance of EIS algorithms. Although these measurements are not absolute since they depend on the sequence being stabilized, and on particular system configura-

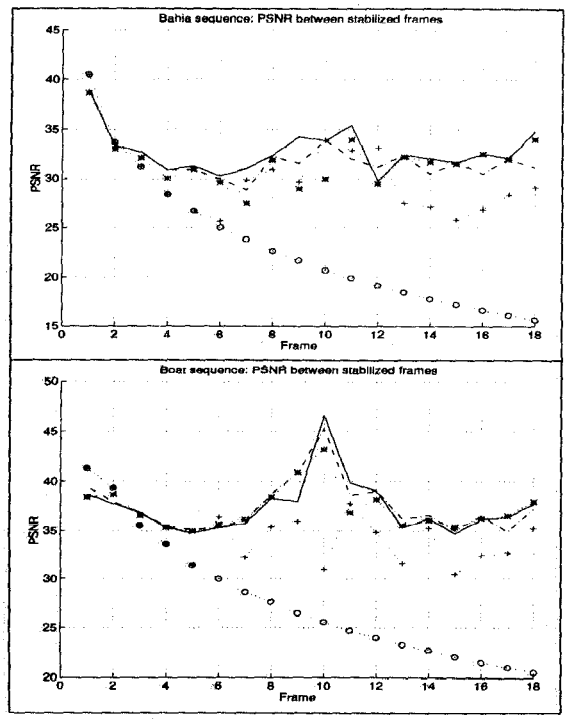

Figure 3: Results from Experiment 2 - determination of minimum translational displacement. The (.*.) curve shows the results for the affine fit, (-) for similarity, (.+.) for Euclidean without subpixel precision, (-.) for Euclidean, and (.o.) for lower bound.

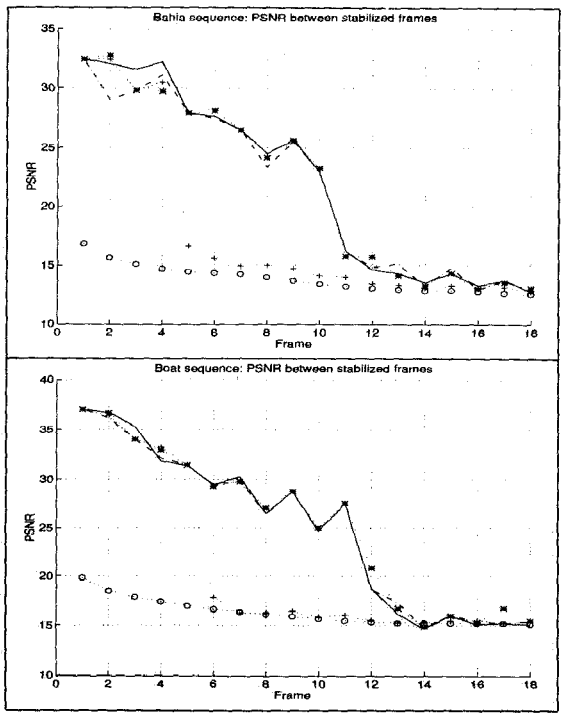

Figure 4: Results from Experiment 3 - determination of maximum translational displacement. The (.*.) curve shows the results for the affine fit, (-) for similarity, (.+.) for Euclidean without multiresolution, (-.) for Euclidean, and (.o.) for lower bound.

tions, they can be used to compare different systems, even those based on different transformation models. They can also be used to evaluate other image registration or global motion estimation techniques, or as development tools to evaluate different configurations.

The evaluation procedures require a few changes on the image composition module, but do not require calibrated sequences. We have compared the performance of stabilization systems based on three different transformation groups, the Euclidean, similarity, and affine groups, and our experimental results show that applying more complex models to fit the data does not necessarily produce better results. Actually, it turns out that the more complex models are more sensitive to tracking errors, causing them to perform worse than the simpler models.

6. REFERENCES

[1] S. Balakirsky and R. Chellappa. Performance characterization of image stabilization algorithms. Technical Report CAR-TR-822, Center for Automation Research, April 1996.

[2] J.L. Barron, D.J. Fleet, and S.S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994.

[3] P. Burt and P. Anandan. Image stabilization by registration to a reference mosaic. In *Proc. DARPA Image Understanding Workshop*, pages 425–434, Monterey, CA, November 1994.

[4] L.S. Davis, R. Bajcsy, R. Nelson, and M. Herman. RSTA on the move. In *Proc. DARPA Image Understanding Workshop*, pages 435–456, Monterey, CA, November 1994.

[5] Z. Durić and A. Rosenfeld. Stabilization of image sequences. Technical Report CAR-TR-778, Center for Automation Research, University of Maryland, College Park, 1995.

[6] M. Hansen, P. Anandan, K. Dana, G. van der Wal, and P.J. Burt. Real-time scene stabilization and mosaic construction. In *Proc. DARPA Image Understanding Workshop*, pages 457–465, Monterey, CA, November 1994.

[7] M. Irani, B. Rousso, and S. Peleg. Recovery of ego-motion using image stabilization. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 454–460, Seattle, WA, June 1994.

[8] C.H. Morimoto and R. Chellappa. Fast electronic digital image stabilization. In *Proc. International Conference on Pattern Recognition*, Vienna, Austria, August 1996.

[9] C.H. Morimoto and R. Chellappa. Fast 3d stabilization and mosaicking. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Puerto Rico,PR, June 1997.

[10] H. Sawhney, S. Ayer, and M. Gorkani. Model-based 2d and 3d dominant motion estimation for mosaicing and video representation. In *Proc. International Conference on Computer Vision*, pages 583–590, Cambridge, MA, June 1995.

[11] Y.S. Yao, P. Burlina, and R. Chellappa. Electronic image stabilization using multiple visual cues. In *Proc. International Conference on Image Processing*, pages 191–194, Washington, D.C., October 1995.