Recovery Video Stabilization Using MRF-MAP Optimization

Soo Wan Kim, Kwang Moo Yi, Songhwai Oh, and Jin Young Choi
EECS Department, ASRI, Seoul National University, Korea
{*swkim,kmyi*}*@neuro.snu.ac.kr,* {*songhwai,jychoi*}*@snu.ac.kr*

Abstract

In this paper, we propose a novel approach for video stabilization using Markov random field (MRF) modeling and maximum a posteriori (MAP) optimization. We build an MRF model describing a sequence of unstable images and find joint pixel matchings over all image sequences with MAP optimization via Gibbs sampling. The resulting displacements of matched pixels in consecutive frames indicate the camera motion between frames and can be used to remove the camera motion to stabilize image sequences. The proposed method shows robust performance even when a scene has moving foreground objects and brings more accurate stabilization results. The performance of our algorithm is evaluated on outdoor scenes.

1. Introduction

In real world visual surveillance, there exists many factors, e.g., wind, rain, and vibration, which cause poor quality in video sequences. In addition, with a handheld camera or a camera mounted on the mobile platform, it is not easy to construct stable video sequences. These cases cause shaky video sequences and, in worst case, it is impossible to figure out objects clearly in the image, e.g., object shapes can be blurred. Video stabilization is an video enhancement method which can be applied in these situations to enhance corrupted video sequences for high level analyses, such as object detection and recognition.

Video stabilization has been researched thoroughly and a number of approaches have been proposed. One of the proposed methods is applying tracking methods in stabilization. In [1] and [2], authors used particle filters to track the movement of pixels between two frames and estimated the global camera motion. In [3], Battiato used scale invariant feature transform (SIFT) features for tracking-based video stabilization. In his work, he adopted a feature-based motion estimation algorithm. Another SIFT-based approach was proposed by Shen [4], and, in his work, he reduced the dimension of extracted features by principal component analysis. Moreover, transform property-based algorithms were also used to estimate camera motion for stabilization. Hong applied block matching on polar transform [5] and Yan adopted Hilbert Huang Transform to stabilize video sequences [6]. Another way to solve the stabilization problem is local patch finding. Ko et. al suggested a local patch finding method by sliding windows on gray coded bit planes [7]. Lastly, an edge mapping algorithm for motion estimation was proposed by Liu [8].

However, these stabilization algorithms only consider two consecutive frames at a time. Hence, they cannot be applied reliably when a scene has independently moving foreground objects and stabilization accuracy can be low in general. This paper presents a method for solving those remaining problems by formulating the video stabilization problem as MRF-MAP energy minimization over a sequence of frames jointly.

The rest of this paper is organized as follows. In Section 2, the joint video stabilization problem is formulated and experimental results and conclusions are given in Section 3 and 4, respectively.

2. MRF-MAP Joint Video Stabilization

Before explaining the main approaches, we make three assumptions; 1) a camera does not move too much between two frames, 2) a camera does not move too much in one direction, 3) the scene has a large number pixels occupying background than foreground. The first assumption is necessary to prevent an excessively blurred image, in which image features, such as pixel values, edges, and texture, are not reliable for high level analyses. The second assumption is applied to distinguish a moving video sequence from a shaking video sequence. In a moving camera video sequence, a camera moves in one direction, while shaking camera moves left to right and up and down repeatedly maintaining the center in the long run. The third assumption is to guarantee reliable stabilization results. If there are too

1051-4651/10 $26.00 © 2010 IEEE
DOI 10.1109/ICPR.2010.687

many foreground pixels in the scene when a camera is shaking, it is impossible to estimate the camera motion because every foreground pixels in the scene could have their own movements which are not related to the camera motion.

The proposed algorithm targets for the whole video sequence and is composed of four steps. First, it models image sequences by a Markov random field. Then, MAP optimization is performed to find matching pixels over several frames under the MRF framework. Next, a simple clustering method is used to find the largest cluster of pixel shifts in consecutive frames to calculate the global shift between frames while eliminating outliers. Finally, a stabilized video sequence is recovered by applying global shifts back to image sequences. Each process is explained below.

2.1 MRF Formulation

A video sequence is modeled by a Markov random field which is a graph with nodes and indirected edges connecting nodes. By spanning several images of video to an MRF model, every pixel is represented by a node in a graph and connected to a group of pixels in the next frame. From the first assumption, edges between pixels in a pre-defined neighbor window exist but there are no connecting edges between distant pixels. Each edge contains a probability which represents the likelihood between a pair of pixels. One-to-one matching of pixels is preferred and enforced by using a prior term in the MAP formulation. This setting is described in detail in the next section. By modeling video inputs as described above, the MRF model for a given video sequence can be built as Figure 1. Foreground pixels are also modeled using this model, but, by the third assumption, they are minority and can be removed as outliers.

With this MRF model, the video stabilization problem can be reduced to the problem of finding matching pixels over a sequence of frames. Finding the best set of matchings and analyzing these matchings in MRF to describe the camera motion is the main goal of this paper. As explained before, a bipartite matching scheme is enforced as priors of MRF to guarantee that two paths are less likely jointed at a single node.

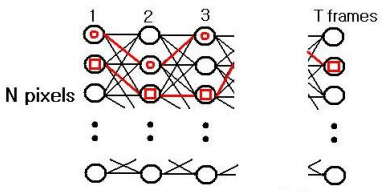


Figure 1. The proposed MRF model:pixel matching;Edges are possible matchings between pixels over consecutive frames and red lines are the matched pairs.

Due to the camera motion, some pixels may have disappeared or a new set of pixels may appear and these pixels cannot be matched to pixels in the previous or next frame. To model those nodes with MRF, null-nodes are added to the graph. In the proposed model, every node is connected to a group of nodes in the next frame and, at the same time, to this null-node with certain probabilities. If connecting to a null-node brings better MAP estimate, the node is connected to the null-node, rather than being connected to a node in the next frame. Many pixels in one frame can be connected to this null-node at the same time; it does not follow the prior constraint.

2.2 MAP Optimization

To stabilize a video sequence, it is necessary to find which pixel at time t is matched to which pixel at time $t+1$ for all t. The proposed algorithm finds pixel matchings based on intensity values in all frames by applying MAP optimization. The probability model of the MRF formulation is shown in (1) below.

$$P(dX|X) \propto P(X|dX)P(dX), \tag{1}$$

where X is a pixel position, and dX is a pixel shift. Here, the likelihood function $P(X|dX)$ can be expressed as

$$P(X|dX) = \Pi_{t,i} P(x_i^t|dx_i^t) \tag{2}$$

with the assumption of independence condition according to time. $P(x_i^t|dx_i^t)$ is calculated among the set $N(x_i^t)$, which is neighborhood of x_i^t in Figure 2. This formulation comes from the first assumption of problem that the camera does not move too much over one frame. The likelihood at time t is defined as

$$P(x_i^t|dx_i^t) = \begin{cases} Z^{-1}e^{-(I(R(x_i^t))-I(x_i^t))^2} & \text{if } R(x_i^t) \in N(x_i^t) \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where $I(x_i^t)$ is the intensity value of x_i^t, $R(x_i^t)$ is a matched pixel of x_i^t at time $t+1$ and Z is a normalizing term. This likelihood function is adopted to represent

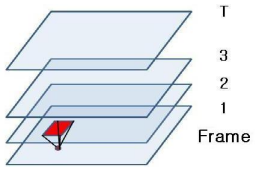


Figure 2. $N(x_i^t)$ used in likelihood

that the pixel with similar intensity in the neighborhood region at time $t+1$ is a reliable match to a pixel at time t. The likelihood of an edge connected to a null-node is pre-defined because a null-node does not have an intensity value. In experiments, it is defined as 0.01 if the maximum of calculated likelihood values in $N(x_i^t)$ is large enough (i.e., if it is likely to be connected to a node in the next frame). Otherwise, it is defined as 0.7 (e.g., it is more likely to be disappeared in the next frame and should be connected to null-node with higher probability).

The prior is,

$$P(dX) = \Pi_t P(dx^t) = \Pi_t \Pi_{i,j,i\neq j} e^{-h(R(x_i)^t), R(x_j)^t))} \tag{4}$$

where $h(y_1, y_2)$ is

$$h(y_1, y_2) = \begin{cases} \beta & \text{if } y_1 = y_2 \\ 1 & \text{otherwise} \end{cases} \tag{5}$$

with β, which is greater than one. This prior makes difficult for two different pixels in the previous frame being matched to a single pixel; the probability will decrease if both of them are assigned to the same pixel.

After defining likelihoods and priors as above, a Gibbs sampling method is applied to solve the MAP problem. In the Gibbs sampling procedure, a configuration of pixel matchings is determined by an iterative algorithm. With posterior probability, the node keeps changing its connecting position in each iterations. With large enough iterations, the MAP solution computed by Gibbs sampling converges to a global solution. This MRF-MAP solution considers all frames jointly, and brings more exact stabilization results.

2.3 Global Shift Calculation

After every pixel movements for all t is known (i.e., when matchings for all nodes are known), the global image shift between t and $t+1$ is calculated. We collect displacements of pixels in the same frame, and use them to compute the global image shift at time t. However, mean value of pixel shifts between two frames is erroneous because some shifts data can be outliers. This can happen when a group of pixels is part of moving objects, which is independent from the camera motion. For this reason, pixels in foreground should be omitted in calculating the global camera shift. Also, pixels connected to null nodes can be outliers. To remove outliers, the proposed algorithm finds the highest density area of data or the largest cluster. The mean shift algorithm with several iterations or K-nearest neighborhood algorithm can be adopted for this purpose. As a result, only pixels inside this high density area are considered in computing global shifts.

2.4 Video Reconstruction

The global transitional shifts of frames at all times are calculated by the previous processes. Because the calculated global shift value at time t indicates how many pixels the image moved or how much a camera moved, reversing pixel shifts can make the video sequences stable. However, some area cannot be recovered when an image is shifted (since there is no information about the shifted area.) The problem of this blank region can be solved by remembering the past scenes and doing image registration as shifting images. For simplicity, it is omitted in this paper.

3 Experimental Results

We tested the proposed algorithm with actual outdoor sequences. They were captured by a hand-held camera and a pole mounted camera which moves unsteadily by winds. There exists foregrounds motions of automobiles in the latter case.

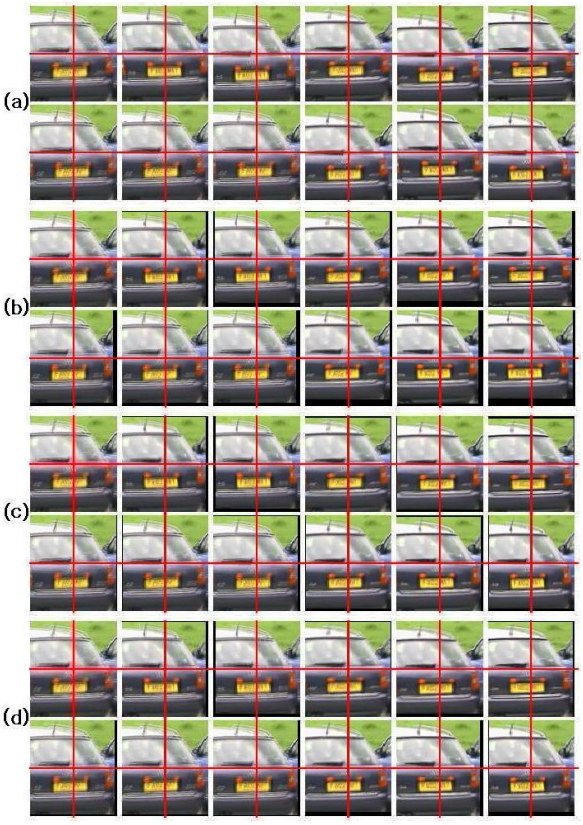


Figure 3. Results: (a) Original video, (b)-(c) Results by [2] and [3], (d) Result by the proposed method.

The resolution of the first test video was 176 by 232 pixels and the results of the proposed algorithm and

other methods are shown in Figure 3. This video is capturing a back of car while camera is hardly shaking. As shown in the Figure 3, the proposed method works better than the other methods. The wiper blades on the back of car looks shaking in the original sequences, while it remains at the same position in the stabilized sequences of the proposed method. The results of other methods looks stable, but are worse than the proposed method. The pixel transitions from ground truths after stabilization algorithms are shown in the Figure 5(a). Original lines describes how the input frames are shaking, and the other lines show the shaking pixel transition distances after [2], [3] and the proposed stabilization algorithms are applied.

The second test video is a surveillance sequence with 120 by 160 pixels and the results are shown in the Figure 4. A camera is mounted on a pole and is shaking and there are several moving foreground objects. In this case, the existence of moving foreground objects lowers the performances of the other algorithms, but the proposed method still works well. The stable position of the yellow sign (where two lines cross) in (d) shows the robust stabilization effect of the proposed method. The pixel transitions from ground truths after stabilization algorithms are illustrated in the Figure 5(b).

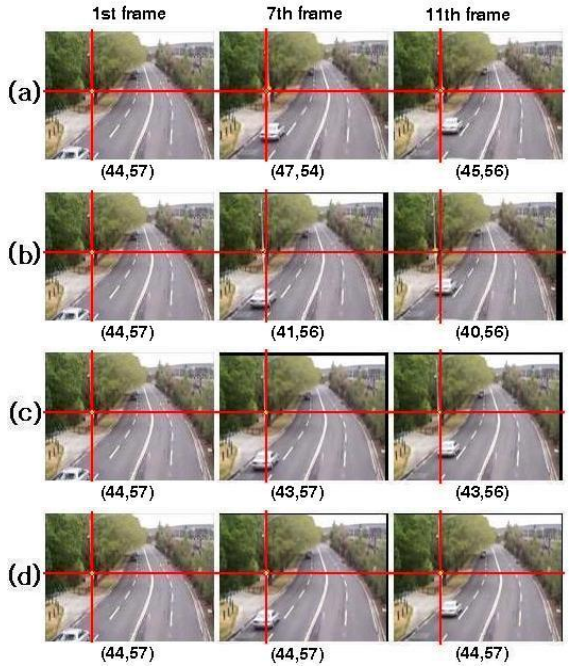
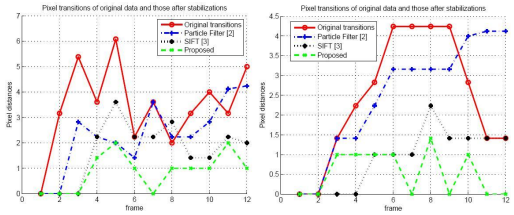


Figure 4. Results: (a) Original image, (b)-(c) Results by [2] and [3], (d) Result by the proposed method, and the position (x,y) of the yellow sign below each image.



(a) Sequences1 (b) Sequences2

Figure 5. Pixel movements:Original,[2],[3] and the proposed method

4. Conclusion

The proposed method is a novel approach for video stabilization. The pixel matching problem based on the MRF model describing video sequences is solved with MAP optimization. The displacements of matched pixels imply shaking camera motion and can be used to eliminate the jitter in camera motion for building stabilized video sequences. Experimental results show robustness and reliability of the proposed algorithm compared to the existing approaches. We plan to improve the performance by developing more sophisticated prior and likelihood models.

References

[1] Junlan Yang, Schonfeld, D., Mohamed, M. Robust Video Stabilization Based on Particle Filter Tracking of Projected Camera Motion, *Circuits and Systems for Video Technology*, vol. 19, issue 7, July 2009, pp 945-954.

[2] Junlan Yang, Schonfeld, D., Chone C., Mohamed, M., Online Video Stabilization Based on Particle Filters, *Image processing 2006*, Oct. 2006, pp 1545-1548

[3] Battiato, S., Gallo, G., Puglishi, G., Scellato, S., SIFT Features Tracking for Video Stabilization, *Image Analysis and Processing 2007*, Sept. 2007, pp 825-830

[4] Yao Shen, Parthasarathy G., Thyagaraju D., Bill P. B., Kameswara R. N., Video Stabilization Using Principal Component Analysis and Scale Invariant Feature Transform in Particle Filter Framework, *IEEE Trans. on Consumer Electronics*, vol. 55, No. 3, August 2009

[5] Hong Z., Changsong D., Junwei L., Fei Y., Ruiming J., Fast digital image stabilization algorithm based on polar transform and circular block matching, *Signal Processing 2008*, Oct. 2008, pp 1124-1127

[6] Hong Z., Changsong D., Junwei L., Fei Y., Ruiming J., Distortion identification technique in video stabilization, *Consumer electronics 2009*, Jan. 2009, pp 1-2

[7] S. Ko, S. Lee, S.Jeon, and E. Kang, Fast digital image stabilizer based on gray-coded bit-plane matching, *IEEE Trans. on Consumer Electronics*, vol. 45, no. 3, pp. 598-603, August 1999.

[8] Liu L., Fu Z., Xie J., Qian W., Edge Mapping: A new Motion Estimation for Video Stabilization, *Computer Science and Computational Technology 2008*, Dec. 2008, pp 440-444