A ROBUST AND NON-ITERATIVE ESTIMATION METHOD OF MULTIPLE 2D MOTIONS

Eun-Young Kang[†], Isaac Cohen[‡], Gérard Medioni[‡]

{elkang, icohen, medioni}@usc.edu

California State University, Los Angeles[†]

University of Southern California[‡]

ABSTRACT

This paper focuses on extracting 2D parametric motion regions from uncalibrated images. Our approach simultaneously infers and detects multiple image regions characterized by 2D motions, affine or homography transformations, from noisy initial matches. This approach is based on: (1) a parametric method to detect and extract 2D affine or homography motion regions; (2) the representation of the matching points in decoupled joint image spaces; (3) the characterization of the property associated with affine transformation in the defined spaces; (4) a non-iterative process to extract multiple 2D motions simultaneously based on tensor-voting; (5) local affine to global homography estimation; (6) region refinement based on a hybrid property: motion and color homogeneity. The robustness of the approach is demonstrated with several results.

1. INTRODUCTION

Grouping based on motion is a key approach for obtaining a layer-based representation of a set of images. In this representation, a layer is defined by a motion descriptor and a compact form of image regions from the original images. These layers can efficiently represent non-rigid objects, transparent object's motion and shallow 3D surfaces. Also, each layer facilitates encoding additional information such as texture and blurring masks.

Over a decade many researchers have popularized the concept and usage of the layered representation based on this grouping concept and have demonstrated promising results [1][6][7][14][15]. However, many methods rely on several assumptions such as pre-specifying the number of layers [2][3][4][14][15], and are sensitive to initial settings or noise. The previous methods characterized as parametric approaches use a specific motion model to define the grouping constraint [6][14]. These methods are iterative and either start with a pre-specified number of layers to fit pixels' motion, or perform repetitive extraction-and-removal of dominant motions. Regularization based methods attempt to apply different smoothing factors and detect accurate layers based on unreliable discontinuity around motion boundaries extracted from locally measured correlation [7]. Approaches based on the use of Markov

Random Fields (MRF) focus on handling discontinuities in the optical flow [4][5]. These methods provide good results, but they rely on a spatial segmentation result in the early stage of algorithm, which may not be practical in many cases. EM-based methods have showed good results but require specifying the number of regions and a significant amount of iterations for reaching satisfactory performance [1][2][15]. When the number of layers is not specified beforehand, an intensive search is required and the optimal grouping of regions is not guaranteed [1]. As an alternative to parametric approaches, non-parametric and non-iterative layer segmentation methods were proposed, which are based on either tensor-voting [11] or normalized graph cuts [12]. Non-parametric approaches attempt to solve general clustering problem. However, applications requiring parameter recovery, such as compression and stabilization, benefit from a motion segmentation approach that groups pixels into regions of similar parametric motions. The subspace constraint analysis of the motion of a planar surface for multiple motions or multiple views is addressed in [9][16]. These approaches successfully analyze the common motion property across multiple motions/frames. However, they are very sensitive to the construction of matrix, which will lead to the extraction of basis of the low dimension, and they require additional iterative grouping processes to cluster pixels after initial segmentation in the low dimension.

Our approach focuses on extracting 2D layers from uncalibrated images and presents a robust 2D layer inference method. Our approach has the following properties: it does not require pre-specifyied number of layers, extracts multiple motion layers non-iteratively based on tensor voting, associates each layer with a parametric 2D transformation, and finally assigns pixels to a layer based on motion saliency and color-region matching.

2. MULTIPLE AFFINE MOTION ESTIMATION

The detection of multiple motion groups with accurate boundaries requires a robust method that identifies different motion groups and assigns image pixels into proper motion groups. We propose a method that uses a

decoupled joint image space to identify affine motion and estimates multiple 2D motions simultaneously using tensor voting. The tensor voting enables us to extract locally salient motion groups while removing local outliers.

2D affine motion is defined by six parameters. The corresponding affine joint image space is defined in the 4D space (x, y, x',y') where (x,y) and (x',y') are matching image points. In the affine model, two equations are independent, and therefore the joint space (x, y, x',y') can be decoupled into two joint spaces (x,y,x') and (x,y,y'). By defining $p_x = (a,b,-1,t_x)^T$ and $p_y = (c,d,-1,t_y)^T$, we have two separate joint spaces $q_x = (x,y,x')^T$ and $q_y = (x,y,y')^T$. We obtain the following equations in decoupled joint image spaces:

$$p_x^T \begin{pmatrix} q_x \\ 1 \end{pmatrix} = ax+by-x'+t_x = 0, \; p_y^T \begin{pmatrix} q_y \\ 1 \end{pmatrix} = cx+dy-y'+t_y = 0$$

These equations define two plane equations in decoupled joint image spaces. A decoupled joint image space is a 3D space and 2D planes in this space represent affine motions. Therefore, by detecting 2D planes from input correspondences, we identify affine motion groups, extract groups of inliers, and estimate affine parameters directly from inliers. In reality, the sparse/dense pixel correspondence contains many mismatches. Therefore, we need to define a robust method for inferring salient plane features within the noisy input correspondences. Our approach uses tensor voting for achieving such selection.

2.1 Tensor voting for affine motion

Tensor voting formalism provides a robust approach for extracting salient structures, such as curve and surface [10]. The input data is encoded as second order symmetric tensors to capture first order differential geometry information and its singularities, and then each input (called *token*) propagates its information to the neighboring data by voting. Voting for neighbourhood is performed through voting fields that are generated based on a scale factor σ (size of neighbourhood). Vote orientation corresponds to the smooth local curve continuation from a voter to a recipient. The extraction of salient structures is inferred from the canonical description of an arbitrary tensor by its eigen-system representing the local geometric properties.

The considered inputs are point correspondences between two images. These correspondences are generated by cross correlation estimation. Then, each correspondence is normalized and is encoded into generic tensor form in 3D. In the 3-D case, a tensor is represented by an ellipsoid. In a more compact form, $S = \lambda_1 \hat{e}_1 \hat{e}_1^T + \lambda_2 \hat{e}_2 \hat{e}_2^T + \lambda_3 \hat{e}_3 \hat{e}_3^T$, where $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$ are the eigenvalues, and $\hat{e}_1, \hat{e}_2, \hat{e}_3$ are the eigenvectors corresponding to $\lambda_1, \lambda_2, \lambda_3$ respectively. The eigenvectors represent the principal directions of the ellipsoid and the eigenvalues encode the size and shape of the ellipsoid. We encode initial correspondences as an ellipsoid with equal eigenvalues for each eigenvector that propagates point information to all direction in any dimension. Once we have all initial tensors, we perform a voting for finding inliers. Due to space constraints, we do not describe the technical details on the plane inference using tensor voting; additional readings on the subject are available [10].

2.2 Multiple motion detection

The motion grouping is performed by clustering points from the most salient points inferred by voting. This is a seed-based region growing method. Our approach does not require specifying the number of motion groups in advance, and the seeds are not selected randomly. The seeds are determined by the inferred saliency from voting process, and their saliencies retain the likelihood measure. The coherence is based on plane-smoothness in the decoupled joint image space and is measured by the angular difference of the normal directions (characterized by the affine coefficients a, b, c, d) and the distance from the space's origin (characterized by the translation parameters Tx and Ty). At each point, the normal orientation of the plane is encoded by the eigenvector e_1 associated to the largest eigenvalue. We compare and cluster neighboring points according to their likelihood. This likelihood measure is provided by the distance functions: $\|d_s - d_i\|^2$, where $d_s = e_{11}^s x_s + e_{12}^s y_s + e_{13}^s x_s'$, $d_i = e_{11}^i x_i + e_{12}^i y_i + e_{13}^i x_i'$, (x_s, y_s, x_s') and (x_i, y_i, x_i') are locations of the seed point, p_s and neighbor point, p_i, and each vector (e_{11}, e_{12}, e_{13}) represents the normal direction of the plane at that location. This function approximates the Euclidean distance between two parameterized motions. If motion difference is smaller than 1 pixel, two points are clustered together. After one region is clustered, the clustering procedure iterates using the next highest salient points. For each clustered motion group, the affine parameters are estimated by analyzing the correlation matrix of the clustered points in each space. The parameters of the affine transform are characterized by the eigenvector associated to the smallest eigenvalue of the correlation matrix.

3. MOTION AND COLOR HOMOGENEITY

We have showed several results for sparse data in [8] that illustrate that our method successfully extracts multiple affine motion layers, while rejecting outliers. Unlike sparse cases, it is required to assign every pixel to a motion group and to extract clean boundaries for dense data. In general, the outliers are present around motion boundaries or homogeneous color regions. Filling holes (or unclustered regions) using the estimated local affine motion does not provide satisfactory results. Indeed, robust motion estimates cannot be inferred in holes nearby motion

boundaries (see Figure 1 (g)(i)). This prevents from clustering these regions based on the motion estimates. To address this problem, we make the following assumption: image pixels in a uniform color region belong to the same motion layer. This assumption is used for refining the extracted motion layers and it provides a complete labeling of image pixels. Our approach performs as follows: we compute a color based segmentation of the reference image and propagate that segmentation to the next image using the computed motion layers associated with affine parameters. We subsequently measure the Sum of Square Difference (SSD) in order to characterize the residual errors generated by mapping color properties of the pixels to its estimated location on the second image. For pixels that were not labeled by the motion layer segmentation, we consider all adjacent layers and estimate the corresponding

SSD residual for the selected pixel. The pixel is then associated to the layer minimizing the SSD residual. The color segmentation used in our approach is a simple threshold based segmentation. Each region tries to fit the motions with respect to the recovered affine motions. Figure 1 (h)(j) and (k) show the refined motion layers that demonstrate visible region boundary enhancement from the initial motion layers. The extracted region boundaries are displayed in Figure 1 (l) in order to visualize the accuracy of the obtained motion · layers. All the segmentation results were produced using the same threshold value. The σ value considered for the voting is 8, and layers were merged when the likelihood was less than one pixel.
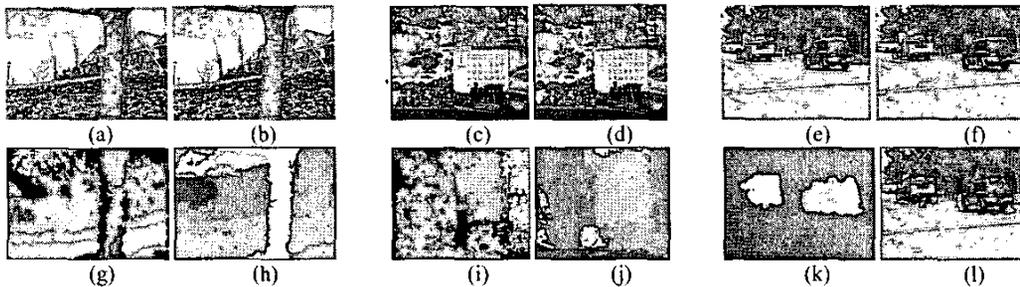

Figure 1 Extraction of multiple 2D affine layers: (a-f) Input frames (g,i) Motion layers (h,j,k) Refined layers (l) Layer boundaries
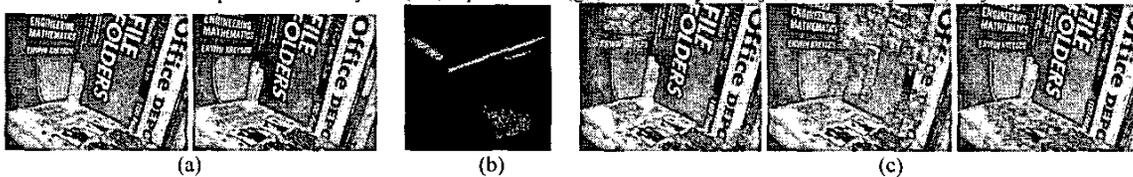

Figure 2 Extraction of multiple 2D homography layers: (a) Inputs (b) Affine patches in 2.5D disparity space with normal directions (c) Detected three homography groups

4. MULTIPLE HOMOGRAPHY ESTIMATION

Affine motion may not be sufficient to approximate segmented motion clusters. Hence, we extend the local affine motion estimation method to 2D homography motion estimation. Images of points on a plane are related to corresponding image points in a second view by a planar homography. In order words, corresponding pixels in two images, which are projections of a world point on a plane in 3D space, are determined by a plane homography.

We group affine patches into the same homography by using plane induced homographies given fundamental matrix F and image correspondences. When two images are rectified using F, each world plane forms a plane in 2.5D disparity space. Also, a motion of a world plane appearing two images is constrained by a homography. Therefore, we estimate unique homography groups by finding unique planes in

2.5D space. To find planes in 2.5D space, fundamental matrix F is computed from locally extracted salient matches. F is computed by least square method. We emphasize here that the matches used for estimating fundamental matrix are locally salient inliers only, and therefore F matrix estimation step is more stable. Given the estimated F matrix we compute rectification homography. A fixed number of epipolar lines are estimated from the fundamental matrix. These epipolar lines are used to rectify the images. After computing the rectification homography, we compute a normal direction for each affine patch. The computation of normal directions is based on the extracted affine parameters and rectification homography parameters. Here, we do not re-compute matches. When a space is defined by $(x, y, f(x,y))$, the direction of normal for a point in this space is defined by $N = (\partial f(x,y)/\partial x, \partial f(x,y)/\partial y, -1)^T$. When images are

rectified with respect to the camera motion (expressed in F matrix), the normal direction of rectified affine patches in the disparity space leads to a homography group. The computation of the normal direction of a plane in 2.5D is dependent on the $f(x,y)$, which defines disparity function. In our case, it is derived from extracted affine parameters and rectification homography parameters. After each normal direction for each patch is computed, the grouping process is performed. If each estimated normal direction is similar (threshold value is 10°), two patches are grouped into a single homography, which corresponds to a world scene plane or planes forming the same angle with the image plane.

Let us consider images containing three planar objects as in Figure 2 (a). Our approach extracted 33 local affine groups from sparse initial matches. Based on matches from extracted affine groups, fundamental matrix F and rectification homographies are computed. Then, for each affine patch, the new normal direction of the patches is computed by using rectification homography and affine motion parameters. Figure 2 (b) shows the plotted normal directions in 2.5D disparity space. Clearly, only three distinctive normal directions are observed, and these three normal correspond to three world planes (books) in the original input images. We can also observe 33 affine groups belong to one of these groups. Figure 2 (c) shows grouping results. Given 10° degree threshold value, 3 homography groups are successfully detected. Note that in the first image in Figure 2 (c), the method successfully detects that the side of 'office-depot' box has same motion as the homography group of 'math' book in the left.

5. CONCLUSION

This paper presented a method to infer 2D parametric layers from pair of images. The robust processing method analyzes the 2D motions, affine and homography, from a video and represents consistent motions into layers. The algorithmic complexity of the described approach is analyzed by subtasks: the most time consuming task is the tensor voting whose complexity is $O(nM)$, where n is the number of pixels and M is the average number of pixels around each pixel. This depends on sigma value considered. We use an 8x8 or 16x16 window size. Currently, our method is relatively slow. However, a coarse-to-fine implementation could improve the performance. In the future, further evaluation needs to be performed along with a quantitative comparison to previous approaches, such as EM-based methods. Possible extensions of the method are: the refinement of the motion boundaries near occlusion areas, the integration of several image

properties, such as motion, color and edge, in the tensor voting framework, as well as the extension to multiple frames is required in order to achieve a true layered representation of a video stream.

7. REFERENCES

[1] S. Ayer et al., Layered Representation of Motion Video, *ICCV*, 1995, pp. 777-784.
[2] S. Baker et al., A Layered Approach to Stereo Reconstruction, CVPR, 1998.
[3] M. Ben-Ezra, S. Peleg and M. Werman, Efficient Computation of the Most Probable Motion from Fuzzy Correspondences, IEEE Workshop on Applications of Computer Vision (WACV), 1998.
[4] Y. Boykov, O. Veksler, R. Zabih, Markov Random Fields with Efficient Aproximations, CVPR, pp. 648-655, 1998.
[5] M. Gelgon and P. Bouthemy, A Region-Level Graph Labelling Approach to Motion-Based Segmentation, CVPR, pp. 514-519, 1997.
[6] M. Irani and S. Peleg, Motion Analysis for Image Enhancement: Resolution, Occlusion, and Transparency, Journal of Visual Communication and Image Representation (VCIR), Vol. 4, No. 4, pp. 324-335, December 1993.
[7] S. X. Ju, M. J. Black and A. Jepson, Skin and Bones: Multi-layer, Locally Affine, Optical Flow and Regularizaton with Transpanrency, Proceedings of CVPR, 1996.
[8] E.Y. Kang, I. Cohen and G. Medioni, Robust Affine Motion Estimation in Joint Image Space using Tensor Voting, International Conference on Pattern Recognition (ICPR), 2002.
[9] Q. Ke and T. Kanade, A Robust Subspace Approach to Layer Extraction, IEEE Workshop on Motion and Video Computing, Orlando, Florida, Dec. 2002.
[10] G. Medioni, et al., A Computational Framework for Feature Extraction and Segmentation, Elsevier, 2000.
[11] M. Nicolescu and G. Medioni, Perceptual Grouping from Motion Cues Using Tensor Voting in 4-D, ECCV, 2000, vol. III, Copenhagen, Denmark, May 2002, pp. 423-437.
[12] J. Shi et al, Motion Segmentation and Tracking Using Normalized Cuts, ICCV, 1998.
[13] A. P. Torr, R. Szeliski and P. Anandan, An Integrated Bayesian Approach to Layer Extraction from Image Sequences, IEEE Trans. on PAMI, 23(3), March 2001, pp. 297-303.
[14] J. A Wang and E. Adelson, Representing Moving Images with Layers, IEEE Transaction on Image Processing Special Issue: Image Sequence Compression, 3(5), September 1994.
[15] Y. Weiss, Smoothness in Layers: Motion segmentation using nonparametric mixture estimation, ", CVPR, 1997,.
[16] L. Zelnik-Manor and M. Irani, Multi-Frame Estimation of Planar Motion, IEEE Trans. on PAMI, Vol. 22, No. 10, October 2000.