PARAMETRIC MOTION MODELING BASED ON TRILINEAR CONSTRAINTS FOR OBJECT-BASED VIDEO COMPRESSION

Zhaohui Sun and A. Murat Tekalp

Department of Electrical Engineering and Center for Electronic Imaging Systems
University of Rochester, Rochester, NY 14627-0126

ABSTRACT

We propose a new parametric motion model based on the so-called "trifocal tensor" representation, which captures rigid 3D motion of static scenes with a depth of field. The estimation of trifocal tensor requires solution of a set of linear equations given at least seven point correspondences across three frames. The proposed parametric representation, called the trilinear model, is superior to other forms such as translational, affine, perspective, and bilinear models, because it can implicitly encode the depth of the scene and 3D motion of the scene/camera under perspective projection unlike others. A video object can thus be represented by its first VOP, a set of trifocal tensors and the corresponding prediction residues. Motion estimation and compensation based on the new parametric model are incorporated into the MPEG-4 Video Verification Model to compare its efficacy for object-based video compression with the state-of-the art motion compensation methods. Experimental results are provided to demonstrate the performance of the trilinear model for object-based video compression.

1. INTRODUCTION

Commonly used 2D parametric motion models, such as affine, perspective, bilinear, cannot accurately represent the projected 3D rigid motion of arbitrary scenes with depth of field. The 2D translational motion model can just handle translations within the image plane. Affine and perspective models only capture 3D rigid motion of a planar object under orthographic and perspective projections, respectively [1]. Alternatively, the perspective transformation accurately models the projected motion due only to rotation of a camera about its center of projection capturing an arbitrary static 3D scene. Otherwise, these models provide a reasonable approximation to the projected motion field if the depth variation of the 3-D scene is much smaller than its distance from the camera. For unrestricted scene structure and camera motion, a residual motion field will remain after a plane projective (perspective) mapping is applied for motion compensation (with respect to a reference plane in the reference view). This residual motion, which can be decomposed into two components, one due to scene structure and the other due to camera translation, is called planar parallax motion [2]. Clearly, the planar parallax motion analysis does not lead to a single parametric mapping between two views.

In this paper, we propose a single parametric mapping, called the trilinear model, between three views that captures rigid 3D motion of static scenes with a depth of field for motion compensation. The trilinear mapping is derived from the so-called "trifocal tensor" representation [4, 5]. This leads to a novel 2D parametric video representation, which captures scene depth and camera motion by a trifocal tensor (27 parameters per video object) without the need for explicit 3D structure and motion estimation. The trilinear model, introduced in Section 2, provides potential for higher coding efficiency through smaller residual error if there is unnegligible depth of field in the scene. Estimation of trifocal tensor requires solution of a set of linear equations given at least seven point correspondences across three frames [4]. Motion estimation and compensation using the trilinear model is discussed in Section 3. The trilinear motion compensation is implemented within the MPEG-4 Verification Model (VM) to compare its efficacy for object-based video compression with the other state-of-the-art motion compensation methods [3]. Experimental results, in Section 4, demonstrate the superiority of trilinear motion model when there is depth variation or camera translation motion in the scene.

This work is supported in part by a National Science Foundation SIUCRC grant and a New York State Science and Technology Foundation grant to the Center for Electronic Imaging Systems at the University of Rochester.

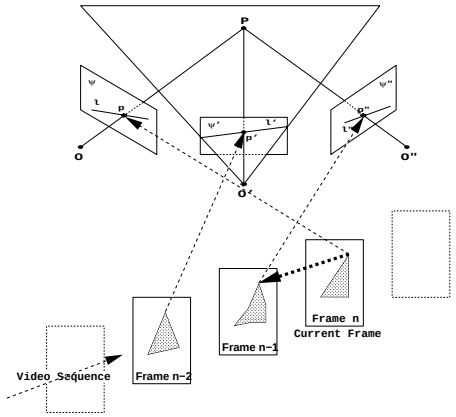

Figure 1: Backward trifocal transfer.

2. TRILINEAR MOTION MODEL

The trifocal tensor extends the notion of the fundamental matrix (epipolar geometry) between two views to three views, and hence implicitly encodes motion and internal parameters of a camera, and all projective geometric constraints across three views, that are independent of scene structure, by 27 parameters [4, 5]. It can be represented by a $3 \times 3 \times 3$ matrix whose entries are given by

$$\alpha_i^{jk} = v'^k b_i^j - v''^j a_i^k \qquad i,j,k = 1,2,3 \tag{1}$$

Interested readers are referred to [4, 5] for details.

Given the trifocal tensor α_i^{jk} across three views ψ, ψ' and ψ'' and the points p and p' in views ψ and ψ' as shown in Fig. 1, the process of finding the corresponding point p'' in view ψ'' is called the trifocal transfer. A line l' in view ψ' together with the optical center O' determines a 3-D plane. A projective line from the first optical center O to an arbitrary point p on the first view ψ intersects the 3-D plane at P. The intersection of the line PO'' with the third view ψ'' determines p'', which is the transfer of the point p onto view ψ''. Thus, the trifocal tensor across three views and the dense motion field between ψ and ψ' are sufficient to perform trifocal transfer.

Shashua [4] shows that a set of three views provides four independent trilinear equations. Given the trifocal tensor α_i^{jk} and the corresponding points p and p', the coordinates of the matching point p'' on the third view ψ'' can be solved uniquely from these equations in the noise free case. Since, we have an overdetermined set of equations, there are various ways of computing p'' using different combination of equations. For example, taking Eqns. (3) and (4) in [4] results in the trilinear mapping

$$x'' = \frac{\alpha_1^{11}x + \alpha_2^{11}y + \alpha_3^{11} - x'(\alpha_1^{31}x + \alpha_2^{31}y + \alpha_3^{31})}{\alpha_1^{13}x + \alpha_2^{13}y + \alpha_3^{13} - x'(\alpha_1^{33}x + \alpha_2^{33}y + \alpha_3^{33})}$$
$$y'' = \frac{\alpha_1^{12}x + \alpha_2^{12}y + \alpha_3^{12} - x'(\alpha_1^{32}x + \alpha_2^{32}y + \alpha_3^{32})}{\alpha_1^{13}x + \alpha_2^{13}y + \alpha_3^{13} - x'(\alpha_1^{33}x + \alpha_2^{33}y + \alpha_3^{33})} \tag{2}$$

The various ways of computing x'' and y'' (different trilinear mapping forms) may result in different trifocal transfers in the case of noisy image correspondences. It is possible to choose the optimal form of the trilinear mapping from a priori knowledge of camera configuration and motion. For example, the mapping (2) is most suitable for dominant motion in the horizontal direction, such as a camera pan, because this mapping is based on a vertical line in the view ψ' (to determine the 3D plane); and thus, does not contain y'. Alternatively, the optimal mapping can be chosen automatically at each pixel based on a line in the view ψ' that passes through p' and is orthogonal to the epipolar line at p'. This, of course, means that the form of the trilinear mapping varies from pixel to pixel, but all such mappings is a function of the same trifocal tensor α_i^{jk}.

3. VIDEO REPRESENTATION AND CODING

This section describes video representation using the trilinear mapping, estimation of the trifocal tensor across three views, and motion compensation and background mosaic generation by the trilinear model.

3.1. Video representation

In MPEG-4, a video sequence is organized into video objects (VO) and video object planes (VOP). A VOP is described by its shape, motion and texture. Based on the trilinear motion model, an N frame VO is represented by its first I-VOP, $(N-1)$ trifocal tensors, and the prediction residues which are DCT coded. For scenes with depth changes which can not be captured by the translational or affine motion model, the trilinear model may result in a smaller prediction residual to be coded, and hence fewer DCT coefficients. If the savings in bit count for coding the residues is more than the cost of coding the trifocal tensors, higher coding efficiency is achieved. This is in general the case when there is a depth field in the scene.

3.2. Motion estimation

Motion estimation in the context of motion compensation by the trilinear model refers to estimating a

dense motion field between successive views and estimating a set of trifocal tensors $\alpha_i^{jk}(n, n-2, n-1)$ for each frame n. The dense motion field can be represented by either a set of block-based translational motion vectors, or a set of global affine parameters $A(n, n-1) = \{a_n, b_n, c_n, d_n, e_n, f_n\}$. The block and affine motion parameters are estimated as described in the MPEG-4 VM [3].

The estimation of the trifocal tensor $\alpha_i^{jk}(n, n-2, n-1)$ across three VOP's follows the following steps:

- Detect a set of "good features" [6] on the first VOP;
- Track the selected feature points to the following frame [6]. Eliminate features which are poorly tracked. Establish feature correspondences across three frames;
- Randomly select 7 point correspondences from this set, and recover the trifocal tensor as described in [7];
- Apply the trilinear mapping to the feature points in frame n to transfer them to frame $(n-1)$. Count the number of features which are successfully reprojected;
- Go to step 3 for a pre-specified number of times and then choose the trifocal tensor which successfully transfers the maximum number of feature points from frame n to $(n-1)$.

For the special case of estimation of the trifocal tensor at the second VOP, a VOP 0 is synthesized by copying VOP 1 to estimate $\alpha_i^{jk}(2, 0, 1)$.

3.3. Motion compensation

The present procedure computes the dense motion field between frames n and $n-2$ using the estimated global affine parameters $A(n, n-1)$ and $A(n-1, n-2)$ as shown in Fig. 1. This choice is motivated by two observations: i) Computation of block-based dense motion vectors is more expensive than estimation of global affine parameters (per object), and ii) Motion compensation by trifocal transfer is not extremely sensitive to correspondence between points p and p'. The procedure is summarized as:

- Scan every pixel p on current VOP in frame n;
- Find the affine warped projection in frame $(n-1)$ by affine transform $A(n, n-1)$ from frame n to frame $(n-1)$. Continue to warp the point to p' in frame $(n-1)$ by the affine transform $A(n-1, n-2)$ from frame $(n-1)$ to frame $(n-2)$. Thus establish point correspondence between p in frame n and p' in frame $(n-2)$.
- Find the projected point p'' in frame $(n-1)$ by the trilinear mapping (2);
- Interpolate the luminance or chrominance value at p'' from its 4 neighbors by bilinear interpolation. Copy the interpolated value to the current frame as the prediction.

The motion compensation is done on both luminance and chrominance channels.

Motion compensation based on trilinear mapping is not perfect due to following reasons: First, there are covered/uncovered regions between three consecutive views. Second, although the trifocal tensors are recovered from the original frames, the current frame is motion compensated from the previous reconstructed frame or sprite, which are blurred and lossy versions of the original frame. Third, the dense motion field from frame n to $n-2$ is approximated by concatenation of two affine transfers, from frame n to $n-1$, then from frame $n-1$ to $n-2$.

3.4. Background mosaic and its update

Background mosaic (sprite) and global affine motion compensation is a standard part of MPEG-4 VM. Background mosaic representation facilitates high efficiency coding [8] for video background. Similar ideas are used to generate and update sprite, using trifocal transfer instead of affine transfer.

4. EXPERIMENTAL RESULTS

Motion compensation by using the trilinear model has been implemented within the MPEG-4 VM (MoMuSys implementation). The bitstream syntax follow exactly the specifications in MPEG-4 VM except that the coding of trifocal tensors as overhead. At present, we attach the trifocal tensors to bitstream without any compression. Every trifocal tensor needs 26 coefficients (The 27th coefficient is set to be 1). Each coefficient is represented as a 4 bytes float number. Thus, 832 more bits per VOP are added to bitstream to represent trifocal tensor. These coefficients could be further compressed by lossless coding to achieve higher efficiency.

Two coders that employ adaptive switching between trilinear and block (TB) models; and affine and block (AB) models have been compared with coders that employ trifocal-only (TO); affine-only (AO); and block-only (BO) models on the background of the video sequence "Stefan" with CIF format (352×288) from frame 220 to frame 268. Table 1 demonstrates that

better coding performance can be achieved by using the trilinear model than by block-matching when there is enough depth variation and camera translation in the scene. These coders are also tested on the foreground object of the video sequence "Cyclamen" with SIF format (352×240) at various sampling rates. Table 2 shows the coding performances where all 300 frames are used. It is clear that BO coder performs very well in this case. Table 3 shows results where 60 frames are sampled at the rate of one out of every 5 frames, and in this case, the performance of the TO coder catches up with the BO coder. Table 4 shows the case where 30 frames are sampled at the rate of one out of every 10 frames, where TO coder gives the best coding efficiency.

Based on these results, it is clear that trifocal motion model is a powerful model. It is, however, desirable to incorporate various motion models into a coding system such that the coder adapts to scenes by switching to the best motion model at each pixel to yield the minimum coding cost.

5. REFERENCES

[1] A. Murat Tekalp, "Digital Video Processing", Prentice Hall, 1995.

[2] M. Irani, B. Rousso and S. Peleg, "Recovery of Ego-Motion Using Region Alignment", IEEE Trans. PAMI, vol. 19, pp. 268-272, 1997.

[3] T. Sikora, "The MPEG-4 video standard verification model", IEEE Trans. CSVT, vol. 7, pp. 19-31, 1997.

[4] A. Shashua, "Algebraic functions for recognition," IEEE Trans. PAMI, vol. 17, pp. 779-789, 1995.

[5] R. I. Hartley, "Lines and points in three views and the trifocal tensor," IJCV, vol. 22, pp. 125-140, 1997.

[6] C. Tomasi and J. Shi, "Good features to track," CVPR, pp. 593-600, 1994.

[7] P. Torr and A. Zisserman, "Robust parameterization and computation of the trifocal tensor," Image and Vision Computing, Vol. 15, pp. 591-605, 1997.

[8] M.-C. Lee, W. Chen, C. Lin, C. Gu, T. Markoc, S. Zabinsky and R. Szeliski, "A layered video object coding system using sprite and affine motion model," IEEE Trans. CSVT, vol. 7, pp. 130-145, 1997.

Sys	Compression Ratio	Average Bits/ Frame	PSNR on Y Channel	PSNR on U Channel	PSNR on V Channel
TB	30.91	39360	30.81	34.98	34.61
AB	28.18	43169	30.72	34.91	34.62
TO	25.52	47669	30.39	34.48	34.12
AO	24.52	49618	30.45	34.59	34.22
BO	18.43	65990	30.53	34.33	34.06

Table 1: Statistics on "stefan" from frame 220 to 268.

Sys	Compression Ratio	Average Bits/ Frame	PSNR on Y Channel	PSNR on U Channel	PSNR on V Channel
BO	119.48	10181	30.76	33.17	33.86
AB	115.86	10500	30.70	33.18	33.85
TB	103.62	11740	30.58	33.20	33.88
TO	74.29	16375	29.96	32.79	33.25
AO	23.60	51526	29.03	31.16	32.21

Table 2: Comparison statistics on 300 frames of "cyclamen" from frame 0 to 299.

Sys	Compression Ratio	Average Bits/ Frame	PSNR on Y Channel	PSNR on U Channel	PSNR on V Channel
BO	94.99	12805	30.77	33.39	34.21
AB	92.57	13141	30.72	33.34	34.24
TB	77.46	15703	30.59	33.28	34.22
TO	65.23	18647	29.95	33.04	33.80
AO	24.43	49800	29.30	31.99	32.97

Table 3: Comparison statistics on 60 frames of "cyclamen" from frame 0 to 299 sampling one out of every five frames.

Sys	Compression Ratio	Average Bits/ Frame	PSNR on Y Channel	PSNR on U Channel	PSNR on V Channel
TO	49.45	24597	29.88	32.97	33.67
TB	47.38	25669	30.19	32.95	33.91
AB	32.83	37055	29.88	32.56	33.52
BO	29.18	41690	29.79	32.31	33.23
AO	23.49	51793	29.41	32.12	33.10

Table 4: Statistics on 300 frames of "cyclamen" from frame 0 to 299 sampling one out of every ten frames.