Virtual Viewpoint Synthesis in Multiview System Based on Affine Transfer

Aiqiu Zuo, Jason Z. Zhang and Q.M. Jonathan Wu

Vision and Sensing Group
National Research Council Canada

3250 East Mall, Vancouver
BC V6T 1W5, Canada
aiqiu.zuo@nrc.ca

William A. Gruver

Intelligent Robotics & Manufacturing Systems
Laboratory, School of Engineering Science

Simon Fraser University, Burnaby
BC V5A 1S6, Canada

Abstract

This paper presents a novel approach to virtual viewpoint synthesis. In the approach we apply an affine transfer method to accomplish precise and fast stereo matching, which is essential for the ever-proposed view-synthesis algorithms. Only five matched reference points are necessary for the stereo matching of any other points. A virtual viewpoint synthesis model for a four-view system is also proposed to accomplish the synthesis of an arbitrary virtual view within the rectangular region bounded by four actual reference views. Both the position and intensity information of the virtual viewpoint are calculated from the reference views with viewpoint adaptation under this synthesis mode. Experiment results proving the good performance of our approach are presented.

1. Introduction

Multiview video can provide more information about scenes than one view. However, the main difficulty with multiview applications is the enormous mount of data to handle. Virtual viewpoint synthesis or interpolated image synthesis from real views is one of the most efficient approaches to multiview problems [1][2]. Also, virtual view synthesis is very useful to the interactive multimedia applications with viewpoint adaptation [3].

A major problem with virtual viewpoint synthesis is disparity estimation. In a few past years, several approaches have been proposed to deal with it. So far the disparity estimation still remains an open problem. With a coplanar stereo camera geometry, which means the vertical disparity is approximately zero all over the stereo images and the scope of the horizon disparity is known as a prior knowledge, a hierarchical block-matching scheme was developed in [3]. An image alignment method was also introduced for stereo matching in [2]. It can convert stereo image of nonparallel views to virtual coplanar ones, taking advantage of fundamental matrix and epipolar constraints. Dynamic programming techniques have been employed in disparity estimation under several constraints, such as correlation, smoothness, disparity gradient limit, inter-scan line compatibility and continuity [4][5]. Usually these approaches to disparity estimation are not only computationally complex, but also noise sensitive.

In this paper we employ an affine transfer method to do the stereo matching. With only five corresponding reference points of stereo images, the disparity estimation can be performed well. This method can not only decrease the computational complexity but also improve the accuracy of disparity estimation. It is efficient especially in a situation where it is easy to get some reference points and either the structure of the image scene is almost planar or the scene is far away enough from cameras.

The most popular model for virtual image synthesis is object-based [3]. 2-D wire frame and 3-D mesh representation are used to model foreground objects [2][6]. But when the structure of the scene is complex and far from cameras, the object segmentation will become much more difficult.

In this paper, the synthesis model is not based on objects. It is not necessary to separate objects from each other when the structure of the view is almost planar or the scene is far away from cameras. The model only takes into account positions and intensities of both the virtual viewpoint and the actual views.

This paper is organized as follows. In section 2, a stereo matching algorithm based on the affine transfer is described. A model of image synthesis is introduced in section 3. Experimental results are presented in section 4 and conclusion remarks are given in section 5.

2. Stereo Matching Based on Affine Transfer [6]

Given any four non-coplanar points $\overline{P}_i$ (i=0,1,2,3) in P^3, the vectors

$$\overline{\varepsilon}_i = \overline{P}_i - \overline{P}_0 (i = 1,2,3) \qquad (1)$$

form a basis spanning of a 3D linear space. Therefore, any other point $\overline{P}_i \in P^3 (i = 4,5,...)$ can be obtained linearly by

$$\overline{P}_i = \overline{P}_0 + \alpha_i \overline{\varepsilon}_1 + \beta_i \overline{\varepsilon}_2 + \gamma_i \overline{\varepsilon}_3 \qquad (2)$$

where $\alpha_i, \beta_i, \gamma_i$ are the affine coordinates of point $\overline{P}_i$.

The 3D-2D transformation of an affine camera model can be expressed as

$$\overline{p}_n = M\overline{P}_n + \overline{t} \qquad (3)$$

where M is an arbitrary 2×3 matrix and $\overline{t}$ is a 2-vector. Then we have

$$\begin{aligned} \overline{p}_i &= M\overline{P}_i + \overline{t} (i = 4,5,...) \\ \overline{p}_0 &= M\overline{P}_0 + \overline{t} \end{aligned} \qquad (4)$$

By substituting Eq. (4) with Eq. (2), we obtain

$$\overline{p}_i - \overline{p}_0 = \alpha_i \overline{e}_1 + \beta_i \overline{e}_2 + \gamma_i \overline{e}_3 \qquad (5)$$

where $\overline{e}_j = M\varepsilon_j (j = 1,2,3)$.

With the affine epipolar constrains for stereo images [7], we have:

$$ax_i^l + by_i^l + cx_i^r + dy_i^r = -1 \qquad (6)$$

where (x_i^l, y_i^l) and (x_i^r, y_i^r) are image coordinates of pixel i in the left and right images respectively.

From four non-coplanar corresponding points over the stereo images, the four unknowns *a, b, c, d* can be determined, which are the same for any matched points over stereo images and only relevant to the cameras' parameters and poses. With Eq. (5) and Eq. (6), equations in unknowns $\alpha_i, \beta_i, \gamma_i, x_i^l, y_i^l$ (for stereo matching from left to right x_i^r, y_i^r are the unknowns instead of x_i^l, y_i^l) can be expressed as

$$\begin{cases} ax_i^l + by_i^l + cx_i^r + dy_i^r = -1 \\ x_i^l = x_0^l + \alpha_i e_{1x}^l + \beta_i e_{2x}^l + \gamma_i e_{3x}^l \\ y_i^l = y_0^l + \alpha_i e_{1y}^l + \beta_i e_{2y}^l + \gamma_i e_{3y}^l \\ x_i^r = x_0^r + \alpha_i e_{1x}^r + \beta_i e_{2x}^r + \gamma_i e_{3x}^r \\ y_i^r = y_0^r + \alpha_i e_{1y}^r + \beta_i e_{2y}^r + \gamma_i e_{3y}^r \end{cases} \qquad (7)$$

which can be rewritten as:

$$A\overline{x} = \overline{b} \qquad (8)$$

where $\overline{x}' = (\alpha_i, \beta_i, \gamma_i, x_i^l, y_i^l)$;

$$A = \begin{bmatrix} e_{1x}^r & e_{2x}^r & e_{3x}^r & 0 & 0 \\ e_{1y}^r & e_{2y}^r & e_{3y}^r & 0 & 0 \\ e_{1x}^l & e_{2y}^l & e_{3y}^l & -1 & 0 \\ e_{1y}^l & e_{2y}^l & e_{3y}^l & 0 & -1 \\ 0 & 0 & 0 & a & b \end{bmatrix}$$

$$\overline{b} = \begin{bmatrix} x_i^r - x_0^r \\ y_i^r - y_0^r \\ -x_0^l \\ -y_0^l \\ -1 - cx_i^r - dy_i^r \end{bmatrix}$$

Then the stereo matching algorithm is formulated as:

(1) With four matched feature points over stereo images, the parameters of affine epipolar constraint can be determined with Eq. (6);

(2) Also with four corresponding feature points, match every pixel of a source image to a destination image with Eq. (8).

However, the four feature points used in step 1 can not be the same as those used in step 2. Otherwise the matrix A will be singular. So at lest five matched points should be necessary [7].

The algorithm is simple, reliable and robust to noises. Because the matrix A changes with the matched feature points and the affine camera model is employed in this algorithm, the precision of stereo matching depends on the image coordinates of the feature points and the configuration of the system. The more precise the image coordinates of the corresponding feature points and the more consistent the system with the affine camera model, the better the precision of the stereo matching will be.

3. Virtual Viewpoint Synthesis

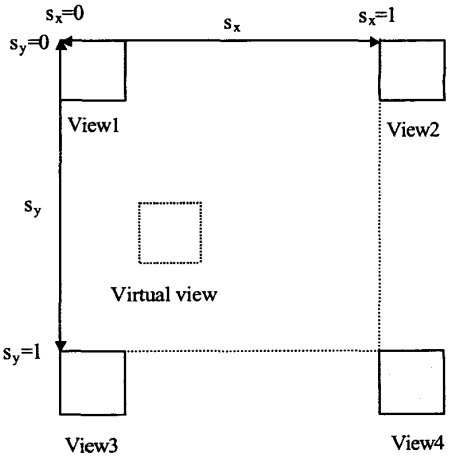


Figure 1 Virtual viewpoint synthesis model for a four-view system

A four-view system is shown in Figure 1, where cameras capture images I_1, I_2, I_3 and I_4 respectively. In this system all actual views are coplanar. We define two posi-

tion parameters, s_x and s_y, to represent the position of viewpoints. The four actual views are positioned at *(0,0), (1,0), (0,1),* and *(1,1)*. With this system configuration, there will be no vertical disparity between I_1 and I_2 and between I_3 and I_4. Also, there will be no horizontal disparity between I_1 and I_3 and between I_2 and I_4. Image at any virtual viewpoint, $(s_x, s_y)(0 \le s_x, s_y \le 1)$, can be interpolated from the four images at the actual views.

With the algorithm described in Section 2, stereo matching can be performed between any pair of images. For pixels of the portion visible in four views, the position in the interpolated image at any virtual viewpoint, (x_v, y_v), can be determined by

$$\begin{aligned} x_v &= x_1(1-s_x) + x_2 s_x \\ y_v &= y_1(1-s_y) + y_3 s_y \end{aligned} \tag{9}$$

where $(x_i, y_i)(i=1,2,3,4)$ are the coordinates of the corresponding pixel in view i. We also have $x_1 = x_3, x_2 = x_4, y_1 = y_2, y_3 = y_4$ because of the special configuration of this multiview system.

The intensity of the pixel, i_v, depends on the intensity measures of the corresponding pixels in images, i_1, i_2, i_3 *and* i_4, and is computed by

$$i_v = w_1 i_1 + w_2 i_2 + w_3 i_3 + w_4 i_4 \tag{10}$$

where w_i $(i=1,2,3,4)$ are the weights for the four real views and determined by

$$\begin{aligned} w_1 &= (1-s_x)(1-s_y) \\ w_2 &= s_x(1-s_y) \\ w_3 &= (1-s_x)s_y \\ w_4 &= s_x s_y \end{aligned} \tag{11}$$

which reasonably satisfy

$$\sum_{n=1}^{4} w_n = 1 \tag{12}$$

$$\begin{cases} s_x = 0, s_y = 0, i_v = i_1 \\ s_x = 1, s_y = 0, i_v = i_2 \\ s_x = 0, s_y = 1, i_v = i_3 \\ s_x = 1, s_y = 1, i_v = i_4 \end{cases} \tag{13}$$

Because in general only a portion of the scene is visible for all four actual views, some parts around the edges of an interpolated image are determined from only two or even one view. For pixels of areas which are only visible in two views, the intensity are determined by

$$i_v = w_m i_m + w_n i_n \tag{14}$$

If $m=1$ and $n=2$ or $m=3$ and $n=4$, which means no vertical disparity between these two views, we define the weights as

$$\begin{aligned} w_m &= s_x \\ w_n &= 1 - s_x \end{aligned} \tag{15}$$

The position is determined only in one direction

$$x_v = x_m(1-s_x) + x_n s_x \tag{16}$$

As to another direction, pixels are only filled in the synthesized image sequentially.

Similarly, if $m=1$ and $n=3$ or $m=2$ and $n=4$, which means no horizontal disparity between two actual views, we define the weights for the intensities in Eq. (14) as

$$\begin{aligned} w_m &= s_y \\ w_n &= 1 - s_y \end{aligned} \tag{17}$$

The position is determined by

$$y_v = y_m(1-s_y) + y_n s_y \tag{18}$$

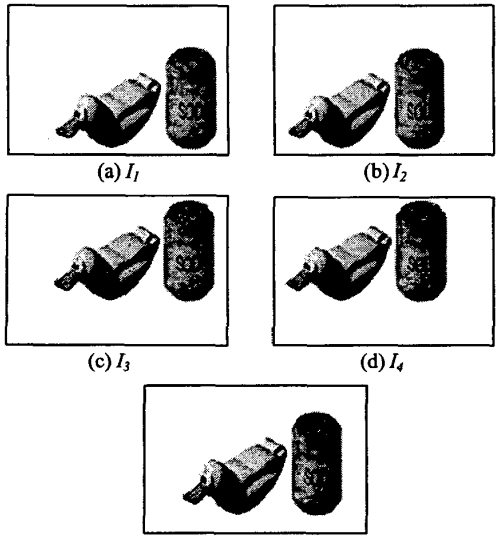
Since those pixels are only visible in only one view, we just fill them in the synthesized images with the pixels sequentially. With this virtual viewpoint synthesis model, the position and intensity of synthesized pixels depend on the location of the virtual viewpoint. Therefore, the interpolated image is obtained with viewpoint adaptation.

4. Experiments

The robustness of the algorithms of the stereo matching and image synthesis described in section 2 and 3 is proved by virtual viewpoint synthesis with two sets of images of four actual views, which are named as DUCK and CROCODILE. DUCK is a set of man-made images using CorelDream3D with an affine camera model. CROCODILE is a set of real images, where the distance between the scene and cameras is about 2 meters and the distances between the cameras along s_x and s_y are about 0.25 meters. The resolution is 246*164 pixels for DUCK images and 130*90 pixels for CROCODILE.

Experiments show that we synthesize virtual viewpoint images with a good quality. The scene in the interpolated image remains clear and the position of the object is just at where it is supposed to be according to the model.

However, compared to those of the actual images, the qualities of the synthesized images degrade a little. Firstly, the image becomes a little blurred. This is because the stereo matching cannot be performed exactly correctly and there are always some perspective effects in the four actual views. Secondly, there are some intensity discontinuities between different parts of the interpolated image in Figure 3 (e). This is because only one part of a scene is visible in all actual images and its view can be determined by the four views; the rest can only be visible in fewer views and the view of these portions are therefore determined by only one or two views. Also since the illuminates are different in different actual views, the synthesized image may show inconsistencies in peripheral parts.



(a) I_1 (b) I_2

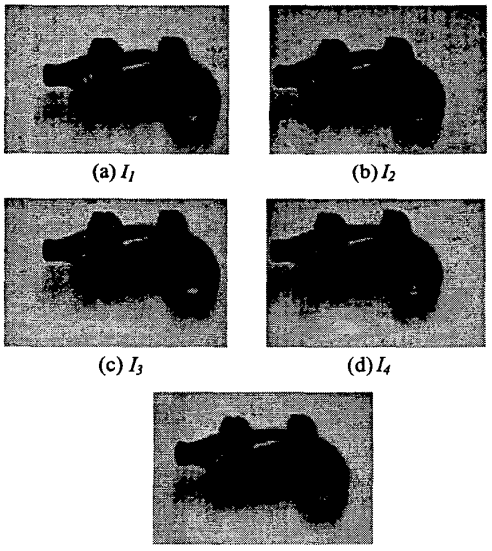
(c) I_3 (d) I_4

(e) Image interpolated (s_x=0.5,s_y=0.5)

Figure 2 DUCK images

(a)~(d) The four actual images

(e) The synthesized image



(a) I_1 (b) I_2

(c) I_3 (d) I_4

(e) Image interpolated (s_x=0.5,s_y=0.5)

Figure 3 CROCODILE images

(a)~(d) The four actual images

(e) The synthesized image

5. Conclusions

In this paper a new method for stereo matching, which is based on an affine transfer method, is presented. Only five matched reference points are necessary for stereo matching. It works well, especially in the situation where either the structure of an object in a scene is planar or the scene is far enough from cameras. A novel model for virtual viewpoint synthesis is also formulated to set up a four-view virtual viewpoint synthesis system. Only the intensities of actual views and the position of the virtual view are needed. Experiments show that good quality interpolated images can be obtained.

Since the system synthesizes virtual images with viewpoint adaptation in 2 dimensions, our method has potentials for the applications of multimedia. Furthermore, with the efficient stereo matching method, the synthesis algorithm is also useful for the multiview video compression [1] and then for multiview applications.

References

[1] Belle L. Tseng, Dimitris Anastassiou. Multiview Video Coding with MPEG-2 Compatibility. IEEE Transactions on Circuit and Systems for Video Technology, Vol.6, No. 4, 1996, pp: 414-419.

[2] Ru-Shang Wang, Yao Wang. Multiview Video Sequence Analysis, Compression, and Virtual Viewpoint Synthesis. IEEE Transactions on Circuit and Systems for Video Technology, Vol.10, No. 3, 1997, pp: 397-410.

[3] Jens-Rainer Ohm, Ebroul Izquierdo M.. An Objected-Based System for Stereoscopic Viewpoint Synthesis. IEEE Transactions on Circuit and Systems for Video Technology, Vol.7, No. 5, 1997, pp: 801-811.

[4] Nikos Grammalidis, Michael G. Strintzis. Disparity and Occlusion Estimation in Multiocular Systems and Their Coding for the Communication of Multiview Image. IEEE Transactions on Circuit and Systems for Video Technology, Vol.8, No. 3, 1997, pp: 328-344.

[5] Y. Ohta, T. Kanade. Stereo by Intra- and Inter-scanline search using dynamic programming. IEEE Transaction on Pattern Anal. Machine Intell., Vol.PAMI-7, pp:139-154, Mar. 1985.

[6] Sotiris Malassiotis, Michael Gerassimos Strintzis. Object-Based Coding of Stereo Image Sequences Using Three-Dimensional Model. IEEE Transactions on Circuit and Systems for Video Technology, Vol.7, No. 6, 1997, pp: 892-905.

[7] Jason Z. Zhang, Q. M. Jonathan Wu, Hung-Tat Tsui, William A. Gruver. Binocular Transfer Methods for Point-Feature Tracking of Image Sequences. IEEE Transaction on Systems, Man, and Cybernetics (submitted).

[8] Richard Hartley, Andrew Zisserman. Multiple View Geometry in Computer Vision. Cambridge University Press, 2000.