Extraction of Camera Parameters Using Projective Invariance for Virtual Studio

SeoWon, Han [*]   JoonWhaon, LEE [*]   Masayuki NAKAJIMA [**]

[*] Graduate School of Engineering of Electronics, *Chonbuk National University KOREA*
[**] Graduate School of Information Science & Engineering, *Tokyo Institute of Technology JAPAN*
hsw@idec.chonbuk.ac.kr

ABSTRACT : Currently virtual studio has used the cromakey method in which an image is captured, and the blue portion of that image is replaced by a graphic image or a real image. The replaced image must be changed according to the camera motion. This paper proposes a novel method to extract camera parameters using the recognition of pentagonal patterns which are painted on the blue screen. The corresponding parameters are position, direction and focal length of the camera in the virtual studio. At first, pentagonal patterns are extracted through image processing, and the matched points of two projected patterns are found using invariant features of the pentagon. Then, the projective transformation of two projected images and the camera parameters are calculated using the matched points. Simulation results indicate that camera parameters are more easily calculated compared to the conventional methods.

1 INTRODUCTION

The cromakey method is a broadcasting technique widely used in television program or general studio. It is an image merging technique in which an image is captured from studio, and the blue portion of that image is replaced by a computer graphics or a real image using analog signal processing. The replaced image has not changed according to the camera motion in studio until now so that the merged image has lack of reality with a limited expressive power. Virtual studio should use the cromakey method in which the blue portion is replaced by 3-dimensional virtual reality reflecting the motion of camera in studio. This paper proposes a novel method to extract camera motion using the recognition of pentagonal patterns which are painted on the blue screen. The camera motion can be represented by seven parameters as follows.

1. The position of camera in studio (X_0, Y_0, Z_0)
2. The direction of camera (α: Pan, β: Tilt, γ: Role)
3. The focal length of camera (f)

Several methods have been proposed to measure the camera parameters in virtual studio. The one employs the optical or mechanical sensors directly mounted on the camera or lense. In this method, however, sensors are noise-sensitive and inaccurate, and they require calibration. In addition to the problems, the sensor occupies large volume so that it may be obstacle to camera-man and it cannot be used for handy camera or soulder camera. The other popular method uses the recognition of mesh patterns composed of two shades of blue color in which two vanishing points of the mesh patterns are changed according to camera motion. In this method, however, it is impossible to measure the parameters when pan or tilt of camera is zero because there exist only one vannishing point[1].

To measure the camera motion, we use pentagonal patterns which have invariant features to the projective transformation of plane. Figure 6 shows blue panel used in virtual studio which consists of pentagons and background composed of two shades of blue color. Figure 8 shows a captured image after a camera motion in the studio. We can measure the camera motion using the disparity between the pentagons of figure 6 and 8. The disparity can be represented by the projective transformation of two planes. To calculate this transformation, the matched points of two images are needed. To search the points, we use pentagons which have two invariant features for the projective transformation of plane. Finally, we derive the seven parameters of camera motion represented in the equations of projection, translation and direction.

This paper is organized as follows. In Section 2 we consider the proposed algorithm for the extraction of camera parameters in detail. Section 3 describes the implementation and the experimental results, and in Section 4 we summarize the conclusion.

2 EXTRACTION OF CAMERA PARAMETERS

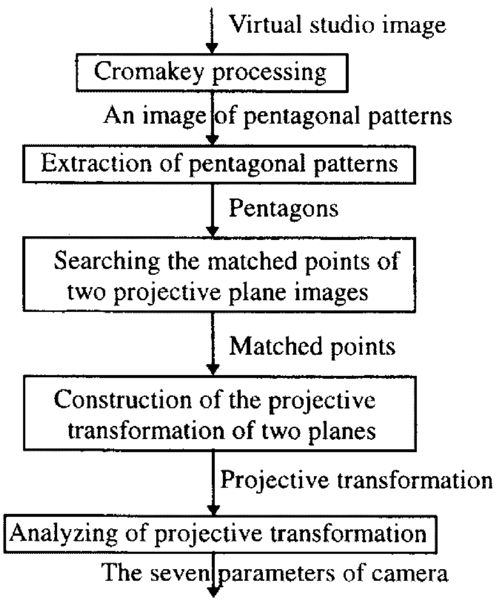


Figure 1. Flow of the proposed algorithm

The proposed algorithm for the extraction of camera parameters consists of following steps.

2.1 Cromakey processing

Virtual studio image, captured in the virtual studio, consists of a blue panel on which the pentagonal patterns are painted some different blue color and characters who act in front of panel (Figure 8). In that image, the portion of pentagonal patterns is used in extracting camera motions in the studio. And, the blue portion except for characters portion is replaced by 3-dimensional virtual reality or video background according to the camera motion. For simplicity, the portion of pentagonal patterns and the characters will now be denoted as the cromakey image and the character image, respectively.

In Figure 2, the color distribution of virtual studio image is divided into *pentagon blue* of the cromakey image, *background blue* of the bluescreen except for pentagons and *character color* of the character image. These distributions have some statistical properties: Character color is far from blue color on hue domain. Also pentagon blue and background blue have blue hues with some difference. Therefore, we can determine whether each pixel of virtual studio image is the pentagon, the background or the character using the color distance to each nominal center of the color distribution. For example, we can calculate the color distances $D_1(\mathbf{X})$ and $D_2(\mathbf{X})$ from a pixel **X** to the centers of pentagon blue and background blue distribution, respectively. And, the pixel **X** can be categorized by following rule.

If $D_1(\mathbf{X}) \le T_1$, then **X** is a pixel of pentagon blue,
else if $D_2(\mathbf{X}) \le T_2$, then **X** is a pixel of background blue,
otherwise, **X** is a character color.

where, T_1, T_2 are the threshold value determined by experiment result.

In this research, we use the Mahalenobis distance measure for the distance between colors as equation (1)[2].

$$D(\mathbf{X}) = (\mathbf{X} - \mu)^T C^{-1} (\mathbf{X} - \mu) \qquad (1)$$

where,

D(**X**) : the Mahalenobis distance from **X** to the sample distribution

X : the color value of pixel (R,G,B)

μ : the mean vector of the sample distribution

C : the covariance matrix of the sample distribution

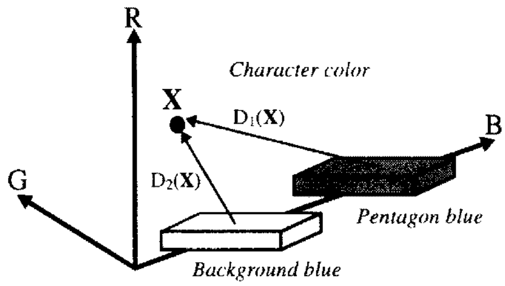


Figure 2. Color distance of studio image

2.2 Extraction of pentagonal patterns

The binarized cromakey image obtained from previous step includes the pentagonal patterns corrupted by illumination and a part of character that is miscategorized in the cromakey processing. In this step, we remove these noises and obtain accurate pentagons using the binary morphology filter. A pentagon can be represented by five vertices or five lines. But, the extraction of vertices is more complicated problem because noise effect is stronger to find vertices. Therefore, we first find the five lines and calculate their intersection points. For extracting the lines, we remove noises using morphological opening operation as in equation (2).

$$A \circ B = (A \oplus B) \ominus B \qquad (2)$$

where, A : the binarized cromakey image

B : the N X N circle mask, structuring element

$\oplus$: the morphological dilation operation

$\ominus$: the morphological erosion operation

The opening operation generally smooths the vertex of pentagon and the line of contour, and eliminates the isolated portion which is smaller than mask size N. N is selected with considering the size of pentagons and the complexity of computation[3]. Figure 3(a) shows the cromakey image, and figure 3(b) shows the opening processed image of figure 3(a) using 10×10 circle mask. After opening, we extract the boundary image using Pitas's method[4], obtain the lines using the variance of tangential coefficient for each pixel. Figure 3(c) shows the extracted lines and vertices of figure 3(b).

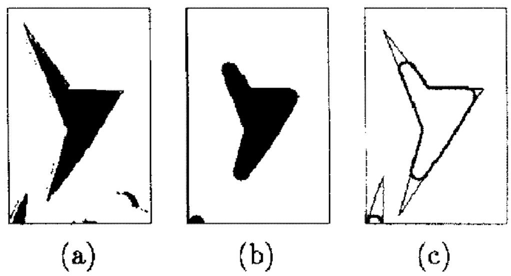


Figure 3. Extraction of pentagons

2.3 Searching the matched points of two projective plane images

Camera parameters defined by camera motion can be calculated by the relationship of two projected plane images. The second image after moving is changed form of the first reference image, according to the camera parameters. For finding the relationship, we must find the correspondences of matched points in two images. Figure 3 shows two projected images of the same plane at different views. Here, the ratio of distance on a line and the ratio of area are not preserved under projective transform of plane, but the two functionally independent invariants defined by equation (3) can be constructed with the five points, $\mathbf{p}_i$, in a plane no three of which are collinear[4]. In this step, a matched pentagon is determined, and then the five matched points are found from a pentagon.

$$\left(\frac{|\mathbf{M}_{431}||\mathbf{M}_{521}|}{|\mathbf{M}_{421}||\mathbf{M}_{531}|},\frac{|\mathbf{M}_{421}||\mathbf{M}_{532}|}{|\mathbf{M}_{432}||\mathbf{M}_{521}|}\right)=\left(\frac{|\mathbf{m}_{431}||\mathbf{m}_{521}|}{|\mathbf{m}_{421}||\mathbf{m}_{531}|},\frac{|\mathbf{m}_{421}||\mathbf{m}_{532}|}{|\mathbf{m}_{432}||\mathbf{m}_{521}|}\right) \quad (3)$$

where, $\mathbf{M}_{ijk}:(\mathbf{P}_i,\mathbf{P}_j,\mathbf{P}_k),\mathbf{P}_i=(X_i,Y_i,1)^T$

$\mathbf{m}_{ijk}:(\mathbf{p}_i,\mathbf{p}_j,\mathbf{p}_k),\mathbf{p}_i=(x_i,y_i,1)^T$

$|\bullet|$: the determinent of $\bullet$

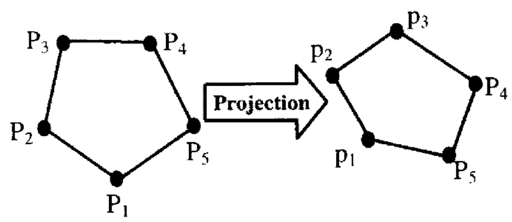


Figure 4. The invariant features of pentagon

2.4 Construction of the projective transformation of two planes

The projected plane is a mathematical concept intended to model the geometric properties of a sequence of one or more perspective projections. A projective transformation between two projected planes can be represented by a generalized linear transformation which is shown in equation (4) and (5)[5].

$$\begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} = \begin{pmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & t_{33} \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \\ 1 \end{pmatrix} \quad (4)$$

$$x_2 = \frac{x_0}{z_0},\ y_2 = \frac{y_0}{z_0} \quad (5)$$

where, (x_1, y_1) is a mathced point of (x_2, y_2) on the projected planes.

The projective transformation matrix requires eight independent parameters to define a unique mapping. Since each point in the plane provides two Cartesian coordinate equations, it is necessary to find four point correspondences between two projectively transformed planes. But, we have more than four points available so that the least-squares method is used to maximally satisfy the conditions.

2.5 Analyzing of projective transformation

The objective of camera calibration is to estimate the internal and external parameters of camera. The first set of parameters represents camera distortion (radial, decentering, thin prism) and camera focal length, and the second set of parameters is the position and direction of camera[6]. In this paper we consider only the seven parameters including the position, the direction of camera and the focal length under no distortion on camera lense. Figure 4 shows camera model where plane is projected on image according to seven paraneters. Here, Image 1 and Image 2 are the projected images of plane 1 under focal length 1, and Image 2 under focal length 2, respectively. Also, plane 2 is the transformed and rotated plane of plane 1 according to **R** and **T** in 3-dimension space.

$$(x_1, y_1) = \left(f_1\frac{X_1}{Z_1}, f_1\frac{Y_1}{Z_1}\right)$$
$$(x_2, y_2) = \left(f_2\frac{X_2}{Z_2}, f_2\frac{Y_2}{Z_2}\right) \quad (6)$$

$$(X_2 \quad Y_2 \quad Z_2)^T = \mathbf{R}(X_1 \quad Y_1 \quad Z_1)^T + \mathbf{T} \qquad (7)$$

where, $\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$ is orthonomal direction matrix, and $\mathbf{T} = (X_0 \quad Y_0 \quad Z_0)^T$ is translation matrix.

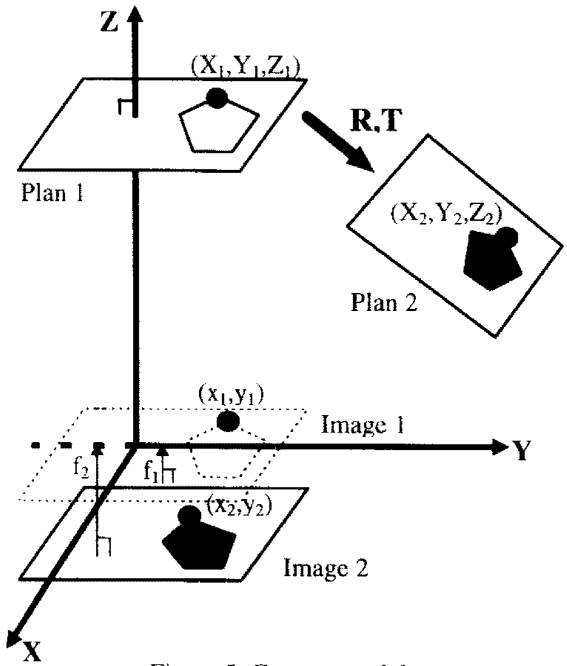


Figure 5. Camera model

$$x_2 = \frac{t_{11}x_1 + t_{12}y_1 + t_{13}}{t_{31}x_1 + t_{32}y_1 + 1}$$
$$y_2 = \frac{t_{21}x_1 + t_{22}y_1 + t_{23}}{t_{31}x_1 + t_{32}y_1 + 1} \qquad (8)$$

$$x_2 = f_2 \frac{r_{11}\frac{x_1}{f_1} + r_{12}\frac{y_1}{f_1} + r_{13}z_1 + X_0}{r_{31}\frac{x_1}{f_1} + r_{32}\frac{y_1}{f_1} + r_{33}z_1 + Z_0}$$
$$y_2 = f_2 \frac{r_{12}\frac{x_1}{f_1} + r_{22}\frac{y_1}{f_1} + r_{23}z_1 + Y_0}{r_{31}\frac{x_1}{f_1} + r_{32}\frac{y_1}{f_1} + r_{33}z_1 + Z_0} \qquad (9)$$

The projection equation (6) and the coordinate transformation (7) can be constructed from the camera model. Also, the equation (9) is the merged form of equation (6) and (7). The equations of projective transformation (4) and (5) can be represented by a single equation (8). Finally, we can extract the camera parameters, the direction **R**, the ratio of focal length f_2/f_1 and the ratio of translation (X_0/Z_1, Y_0/Z_1, Z_0/Z_1) using equation (8) and (9).

3 EXPERIMENTAL RESULTS

In this section we implement the proposed algorithm on six color images captured under different camera parameters in virtual studio. Figure 6 shows the reference image of the camera motion, and figure 8(a) and 8(b) shows the example images after moving camera to the right and to the left respectively in studio. Since the characters can ovelap the pentagons, the number of pentagons should be more than one. Figure 9(a) shows the processed cromakey image of figure 8(b). Also, figure 9(b) shows the extracted pentagons using binary morphology in which the noise of character and the effect of illumination were removed. As we can see, pentagon (1) and (4) overlapped by the character are excluded in finding matched point.

Table 1 shows the extracted seven camera parameters of figure 8. Figure 10 shows the merged images of figure 8 and a real image of figure 7 according to the extracted camera parameters in table 1.

Table 1. Camera parameters

		(a)	(b)
Direction	Pan	22.70°	-14.95°
	Tilt	-5.06°	-5.05°
	Role	-1.24°	-1.20°
Translation	X_0/Z_1	0.328	-0.250
	Y_0/Z_1	-0.123	-0.038
	Z_0/Z_1	0.089	-0.097
Focus	f_2/f_1	0.732	0.765

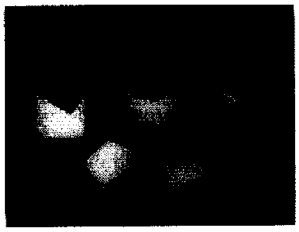
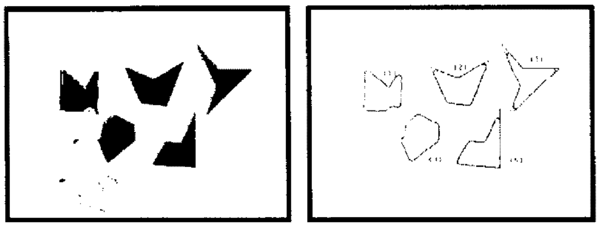


Figure 6. Reference image

Figure 7. Background image

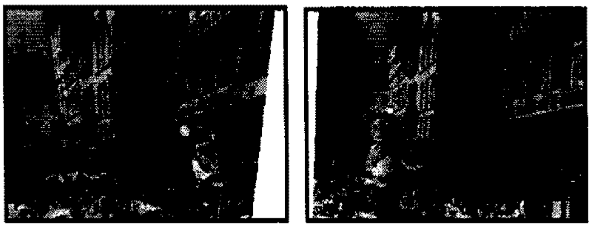
(a) (b)

Figure 8. Virtual studio images



(a) (b)

Figure 9. The extraction of pentagons



(a) (b)

Figure 10. Merged images with figure 7 and figure 8

4 CONCLUSION

We proposed a new method to extract camera parameters using the recognition of pentagonal patterns in the virtual studio. Pentagonal patterns can be extracted from studio image using cromkeying and morphological image processing which can be easily implemented by hardware. Also, using the invariant features of pentagons, we find the matched points of two projected plane images. Finally, the seven parameters are calculated from projective transformation of plane defined by matched points.

In this paper we use only front blue panel. If we add to use more than front panel such as left, right, and bottom panels, we can expect to derive seven camera parameters more exactly. Since in the proposed algorithm the half of processing time is taken in image processing, we will consider to use new patterns such as ellipses that can be easily detected.

REFERENCE

[1] Mikael Tamir: The Orad Virtual Set, International Broadcast Engineering, March, pp.16-24,1996.

[2] J.T. Tou and R.C. Gonzalez: Pattern Recognition Principles, Addision-Wesley, 1974.

[3] R.C. Gonzalez and R.E. Woods: Digital Image Processing, Addision-Wesley, pp.518-560,1992.

[4] Azriel Rozenfield and A.C. Kak: Digital Picture Processing, Vol I,II, 1980.

[5] Joseph L.Mudy and Andrew Zisserman: Geometric Invariance in Computer Vision, MIT Press, pp.463-519, 1992.

[6] Oliver Faugeras: Three-Dimensional Computer Vision, MIT Press, 1993