

# Chroma Key Using a Checker Pattern Background

Hiroki AGATA<sup>†</sup>, *Nonmember*, Atsushi YAMASHITA<sup>†\*a)</sup>, and Toru KANEKO<sup>†</sup>, *Members*

**SUMMARY** In this paper, we propose a new region extraction method using chroma key with a two-tone checker pattern background. The method solves the problem in conventional chroma key techniques that foreground objects become transparent if their colors are the same as the background color. The method utilizes the adjacency condition between two-tone regions of the background and the geometrical information of the background grid line. The procedure of the proposed method consists of four steps: 1) background color extraction, 2) background grid line extraction, 3) foreground extraction, and 4) image composition. As to background color extraction, a color space approach is used. As to background grid line extraction, it is difficult to extract background grid line by a color space approach because the color of this region may be a composite of two background colors and different from them. Therefore, the background grid line is extracted from adjacency conditions between two background colors. As to foreground extraction, the boundary between the foreground and the background is detected to recheck the foreground region whose color is same as the background, and the background region whose color is same as the foreground. To detect regions whose colors are same as the background, the adjacency conditions with the background grid line are utilized. As to image composition, the process that smoothes the color of the foreground's boundary against the new background is carried out to create natural images. Experimental results show that the foreground objects can be segmented exactly from the background regardless of the colors of the foreground objects.

**key words:** chroma key, checker pattern background, region extraction, image composition

## 1. Introduction

Image composition [1] is very important to creative designs such as cinema films, magazine covers, promotion videos, and so on. This technique can combine images of actors or actresses in a studio and those of scenery taken in other places. Robust methods are needed especially for live programs on TV [2], [3].

To perform image composition, objects of interest must be segmented from images, and there are many studies about image segmentation [4], [5], e.g., pixel-based, area-based, edge-based, and physics-based ones. For example, Snakes [6] was proposed as an effective

technique based on edge detection. However, there have not been developed practical methods which are accurate and automatic, while a method with high accuracy is proposed which is realized by human assistance [7]. Qian *et al.* proposed an algorithm that classifies the pixels in an input image into foreground and background based on the color difference between the input image and a pre-recorded background image [8]. The classification result is obtained by computing a probability function and the result is refined using anisotropic diffusion. However, the algorithm does not work well when foreground objects share regions of similar color and intensity with background. It has also restriction of requiring a stationary camera.

As to camera motion, Shimoda *et al.* proposed a method in which the background image alters accordingly as the foreground image is altered by panning, tilting, zooming and focusing operations of the camera [9]. This method is a fundamental technique of virtual studios [2], [3].

As a method that takes the advantage of three-dimensional information, Kanade *et al.* proposed a stereo machine for video-rate dense depth mapping that has a five-eye camera head handling the distance range of 2 to 15m using 8mm lenses [10]. Kawakita *et al.* proposed the axi-vision camera that has up-ramped and down-ramped intensity-modulated lights with an ultrafast shutter attached to a CCD probe camera [11]. These systems can obtain the ranges from the camera to the objects in the scene and extract the objects from the images by using the range information. However, since these systems consist of special devices, it is difficult for ordinary users to realize image segmentation by employing this information.

Chroma key, which is also referred to as color keying or color-separation overlay, is a well-known image segmentation technique that removes a color from an image to reveal another image behind. Objects segmented from a uniform single color (usually blue or green) background are superimposed electronically to another background. This technique has been used for long years in the TV and the film industries.

In image composition, the color  $I(u, v)$  of a composite image at a pixel  $(u, v)$  is defined as:

$$I(u, v) = \alpha(u, v)F(u, v) + (1 - \alpha(u, v))B(u, v), \quad (1)$$

where  $F(u, v)$  and  $B(u, v)$  are the foreground and the

Manuscript received April 3, 2006.

Manuscript revised July 17, 2006.

<sup>†</sup>The authors are with the Department of Mechanical Engineering, Shizuoka University, Hamamatsu-shi, 432-8561, Japan.

\*Presently, the author is with the Department of Mechanical Engineering, California Institute of Technology, Pasadena, CA 91125, U.S.A.

a) E-mail: yamashita@ieee.org

background color, respectively, and  $\alpha(u, v)$  is the so-called alpha key value at a pixel  $(u, v)$  [1]. The color at a pixel  $(u, v)$  is the same as that of the foreground when  $\alpha(u, v)$  equals to 1, and is the same as that of the background when  $\alpha(u, v)$  equals to 0. In chroma key, it is very important to determine the alpha value exactly. Methods for exact estimation of the alpha value have been proposed in applications of hair extraction, transparent glass segmentation, and so on [12]–[17].

However, conventional chroma key techniques using a monochromatic background have a problem that foreground objects are regarded as the background if their colors are similar to the background color, and the foreground regions of the same color are missing (Fig.1). To solve this problem, Smith and Blinn proposed a blue screen matting method that allows foreground objects to be shot against two backing colors [18]. This method can extract the foreground region whose colors are the same as the background color. This alternating background technique cannot be used for live actors or moving objects because of the requirement for motionlessness within a background alternation period.

In order to solve the above problem, we proposed a method for segmenting objects from a background precisely even if objects have a color similar to the background [19]. In this method, a two-tone stripe background is used (Fig.2). As to foreground extraction, the boundary between the foreground and the background is detected to recheck the foreground region whose color is same as the background. To detect the region whose color is same as the background, the method employs the condition that the striped region endpoints touch the foreground contour. If the foreground object has the same color as the background and has parallel contours with the background stripes, endpoints of the striped region do not touch the foreground contour. Therefore, it is difficult to extract such foreground objects (Fig.2).

In this paper, we propose a new method for extracting foreground objects with arbitrary shape in any color by using a two-tone checker pattern background (Fig.3).

## 2. Chroma Key with a Checker Pattern Background

The procedure consists of four steps; background color extraction (Fig.4(a)), background grid line extraction (Fig.4(b), (c)), foreground extraction (Fig.4(d), (e)), and image composition (Fig.4(f)).

### 2.1 Background Color Extraction

Candidate regions for the background are extracted by using a color space approach. Let  $R_1$  and  $R_2$  be the regions whose colors are  $C_1$  and  $C_2$ , respectively, where  $C_1$  and  $C_2$  are the colors of the two-tone background

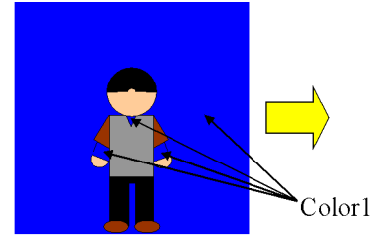


Fig. 1 Chroma key with unicolor background.

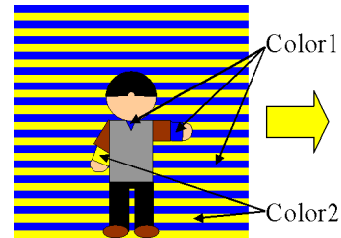


Fig. 2 Chroma key with stripe background.

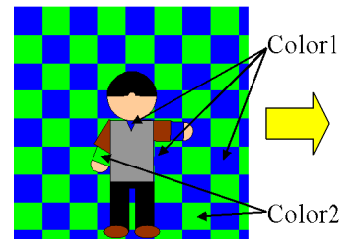


Fig. 3 Chroma key with checker pattern background (proposed method).

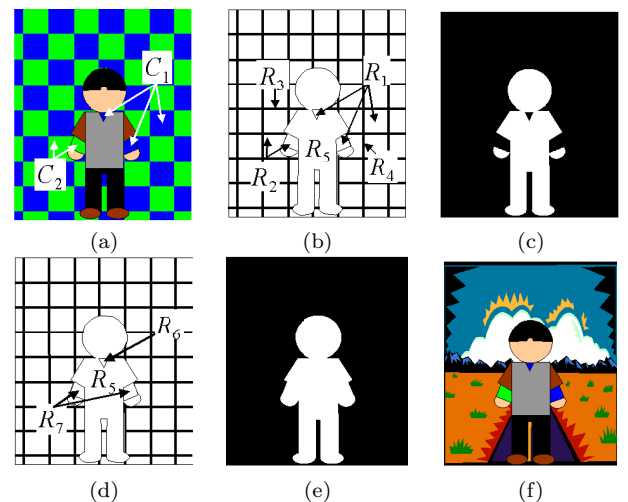


Fig. 4 Region extraction. (a) Original image. (b) Region segmentation without the information of checker pattern. (c) Foreground extraction by the result of (b). (d) Region segmentation using the information of checker pattern. (e) Foreground extraction by the result of (d). (f) Image composition.

in an image captured with a camera (Fig.4(a)). Then region  $R_i$  ( $i = 1, 2$ ) is represented as

$$R_i = \{(u, v) | I(u, v) \in C_i\}, \quad (2)$$

where  $I(u, v)$  is the color of an image at a pixel  $(u, v)$ .

In addition to regions  $R_1$  and  $R_2$ , intermediate grid-line regions between  $R_1$  and  $R_2$  are also candidates for the background. Let such regions be denoted as  $R_3$  and  $R_4$ , where the former corresponds to horizontal grid lines and the other to vertical ones, respectively (Fig.4(b)). The color of  $R_3$  and  $R_4$  may be a composite of  $C_1$  and  $C_2$ , which is different from  $C_1$  and  $C_2$ . Here, let  $C_3$  be the color belonging to regions  $R_3$ ,  $R_4$  or foreground region, and we have the following description:

$$C_3 = \{I | I \notin (C_1 \cup C_2)\}. \quad (3)$$

Figure 5 illustrates a relation among background regions  $R_1$ ,  $R_2$ ,  $R_3$ ,  $R_4$  and pixel colors  $C_1$ ,  $C_2$ , and  $C_3$ .

It is necessary to estimate  $C_1$  and  $C_2$  in individual images automatically to improve the robustness against the change of lighting conditions. We realize this automatic color estimation by investigating the color distributions of the leftmost and rightmost image areas where the foreground objects do not exist as shown in Fig.6(a). The colors in these reference areas are divided into  $C_1$ ,  $C_2$  and  $C_3$  in the HLS color space by

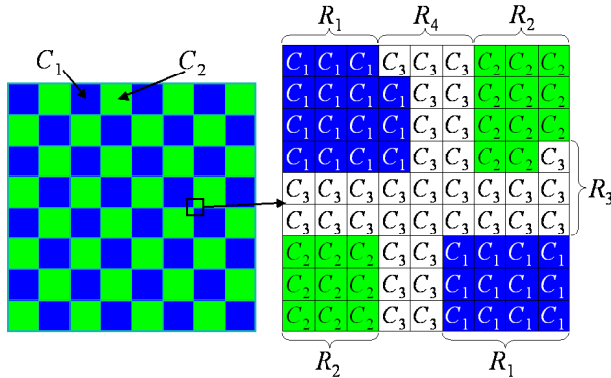


Fig. 5 Background regions and their colors.

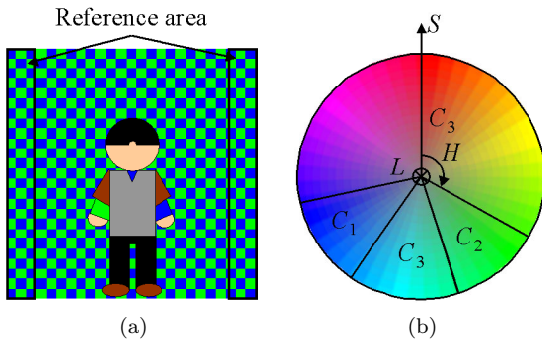


Fig. 6 Background color estimation. (a) Reference areas. (b) HLS color space.

using K-mean clustering. Figure 6(b) shows the distribution of  $C_1$ ,  $C_2$  and  $C_3$  in the HLS color space, where the H value is given by the angular parameter. The HLS color space is utilized because color segmentation in the HLS color space is more robust than in the RGB color space against the change of lighting conditions.

Let  $H_i$  ( $i = 1, 2$ ) be the mean value of the H values of  $C_i$  ( $i = 1, 2$ ) in the reference areas, and let  $h_j$  ( $j = 1, 2, \dots, N$ ) be the H value of each point in the image, where  $N$  is the total number of pixels of the image. Pixels are regarded as background candidate pixels if they satisfy the following condition, where  $T$  is a threshold:

$$|H_i - h_j| \leq T. \quad (4)$$

## 2.2 Background Grid Line Extraction

Background grid lines are extracted by using adjacency conditions between two background colors. Background grid line regions  $R_3$  and  $R_4$  contact with both  $R_1$  and  $R_2$ . The colors of the upper and lower regions of  $R_3$  differ from each other, and also the colors of the left and right regions of  $R_4$  differ from each other. Therefore,  $R_3$  and  $R_4$  are expressed as follows:

$$R_3 = \{(u, v), (u, v + 1), \dots, (u, v + l + 1) | \\ I(u, v + 1) \in C_3, \dots, I(u, v + l) \in C_3, \\ ((I(u, v) \in C_1, I(u, v + l + 1) \in C_2) \text{ or } \\ (I(u, v) \in C_2, I(u, v + l + 1) \in C_1))\}, \quad (5)$$

$$R_4 = \{(u, v), (u + 1, v), \dots, (u + l + 1, v) | \\ I(u + 1, v) \in C_3, \dots, I(u + l, v) \in C_3, \\ ((I(u, v) \in C_1, I(u + l + 1, v) \in C_2) \text{ or } \\ (I(u, v) \in C_2, I(u + l + 1, v) \in C_1))\}, \quad (6)$$

where  $l$  is the total number of the pixels whose color is  $C_3$  in the vertical or horizontal direction.

However, if  $R_3$  and  $R_4$  are included in foreground objects, e.g., when a person wears in part a piece of cloth having the same checker pattern as the background as shown in Fig.7, these regions can not be distinguished whether foreground or background. Therefore, we apply a rule that background grid lines should be elongated from those given in the reference area where foreground objects do not exist. If grid lines in foreground objects are dislocated from background grid lines as shown in Fig.8, they are regarded as foreground regions. Elongation of the background grid lines is realized by the following method.

In the case of horizontal background grid lines, continuous lines exist at the top of the image as shown in Fig.9(a). At any part of the left and right reference area of the image, foreground objects do not exist. Therefore, it is possible to match any horizontal lines between

the left end of the image and the right one by making correspondence from top to bottom one by one. We approximate the horizontal background grid lines behind the foreground object by applying a least mean square method to visible grid-line pairs.

In the case of vertical background grid lines, continuous lines exist at the left and the right end of the image as shown in Fig.9(b). However, if a foreground object is a standing person, it is not always possible to match the vertical lines between the top end of the image and the bottom one. In this case, we estimate the vertical background grid line behind the foreground object by applying a least mean square method only to the line at the top end of the image.

When applying a least mean square method to estimate either horizontal or vertical grid lines mentioned above, we should take into account the influence of camera distortion as a practical problem. Then we should fit higher-order polynomial curves instead of straight

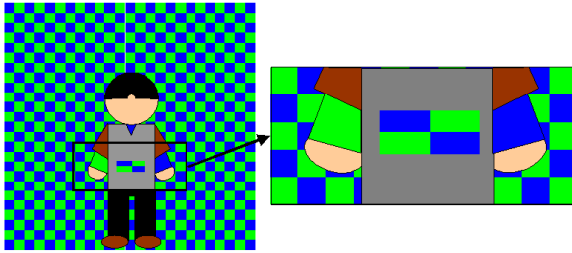


Fig. 7 Checker pattern in the foreground object.

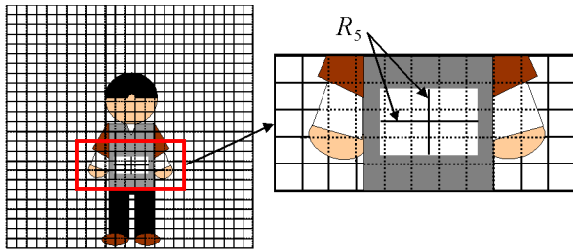


Fig. 8 Background grid line approximation.

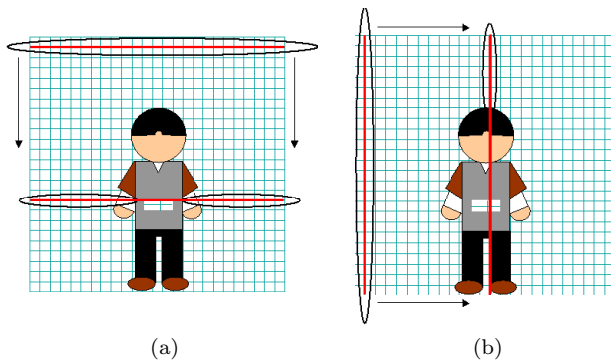


Fig. 9 Background grid line estimation. (a) Horizontal line. (b) Vertical line.

lines to the background grid lines. Here, we may have another approach; i.e. if we use a perfect checker pattern background consisting of squares or rectangles each of which has exactly the same shape, and a distortion-free camera whose image plane is set perfectly parallel to the checker pattern background, the background grid-line extraction procedure will become a very easy one. Comparing to this situation, our procedure seems to be elaborate, but it has such advantages as a camera with an ordinary lens can be used, the camera is allowed to have some tilt, and a checker pattern background which is somewhat distorted is available.

In this stage, all regions whose colors are same as the background are extracted as the background candidates, which may include mis-extracted regions as illustrated in Fig.4(c). Those are regions which belong to the foreground object but have the same color as the background. As shown in Fig.4(d), we define foreground regions whose colors are different from the background as  $R_5$ , and we define mis-extracted regions isolated from the background and neighboring to the background as  $R_6$  and  $R_7$ , respectively. These errors are corrected in the next step.

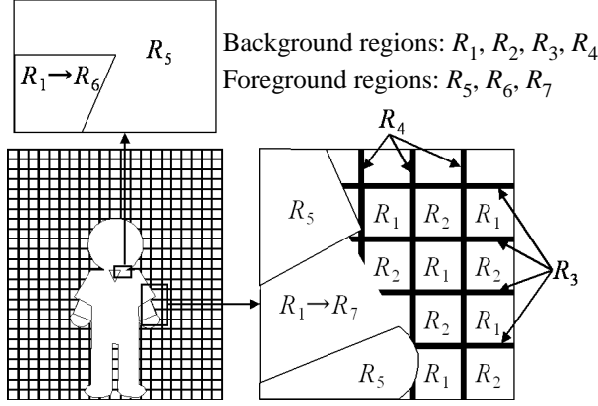
### 2.3 Foreground Extraction

Background candidate regions corresponding to  $R_6$  and  $R_7$  should be reclassified as the foreground, although their colors are same as the background. This reclassification can be realized by adopting the following rules concerning to adjacency with background grid lines.

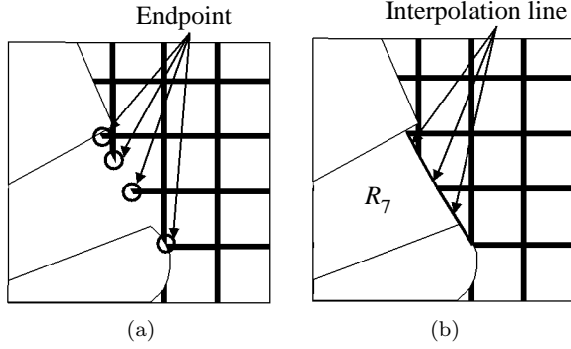
1. If there is a background region candidate which does not connect with the background grid line regions  $R_3$  nor  $R_4$ , it is reclassified as the foreground region  $R_6$  (Fig.10, top).
2. If there is a background region candidate which has an endpoint of a background grid line in its inside, it is divided into two regions; one is a foreground region and the other is a background (Fig.10, right). The dividing boundary of the two regions is given by a series of the interpolation lines each of which is a connection of neighboring background grid-line endpoints (Fig.11(a), (b)). The region containing the background grid line is regarded as the background, and the other is regarded as the foreground region  $R_7$ .

Figure 11 illustrates the above 2nd rule. Background grid-line endpoints shown in Fig.11(a) produce the dividing boundary as a series of interpolation lines as shown in Fig.11(b).

By completing the above procedures, the image is divided into seven regions  $R_1 \sim R_7$ . Regions  $R_5$ ,  $R_6$  and  $R_7$  are the foreground regions (Fig.4(d), (e)). The contours of the foreground objects may not be exact ones, because the interpolation lines do not give fine structure of the contours owing to the simplicity of straight



**Fig. 10** Foreground region whose color is same as the background. Top: Inside the foreground ( $R_1$  is reclassified to  $R_6$ ). Right: Neighboring to the background ( $R_1$  is reclassified to  $R_7$ ).



**Fig. 11** Determination of region  $R_7$ . (a) Background grid line endpoints. (b) Interpolation lines.

line connection. Therefore, we need to execute post processing for contour refinement which is realized by Snakes [6] (Fig.12).

Let  $\mathbf{s}_i = (u, v)$  ( $i = 1, 2, \dots, n$ ) be closed curves on the image plane  $(u, v)$ , and we define Snakes energy as:

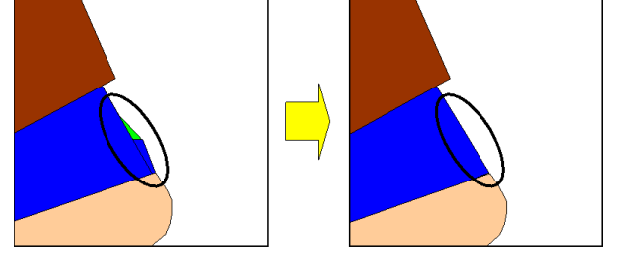
$$E_{snake} = E_{spline}(\mathbf{s}_i) + E_{image}(\mathbf{s}_i) + E_{area}(\mathbf{s}_i), \quad (7)$$

where  $E_{spline}(\mathbf{s}_i)$  is the energy to make the contour model smooth,  $E_{image}(\mathbf{s}_i)$  is the energy to attract the contour model to the edge, and  $E_{area}(\mathbf{s}_i)$  is the energy for the contour model to expand to fit to the reentrant shape [20]. These energies are defined as:

$$E_{spline}(\mathbf{s}_i) = \sum_{i=1}^n (w_{sp1} |\mathbf{s}_i - \mathbf{s}_{i-1}|^2 + w_{sp2} |\mathbf{s}_{i+1} - 2\mathbf{s}_i + \mathbf{s}_{i-1}|^2), \quad (8)$$

$$E_{image}(\mathbf{s}_i) = - \sum_{i=1}^n (w_{image} |\nabla I(\mathbf{s}_i)|), \quad (9)$$

$$E_{area}(\mathbf{s}_i) = \sum_{i=1}^n w_{area} \{u_i(v_{i+1} - v_i)\}$$



**Fig. 12** Region extraction using Snakes.

$$-(u_{i+1} - u_i)v_i\}, \quad (10)$$

where  $w_{sp1}$ ,  $w_{sp2}$ ,  $w_{image}$ ,  $w_{area}$  are weighting factors for each energy.  $I(\mathbf{s}_i)$  is a function of image intensity on  $\mathbf{s}_i$ . Therefore,  $|\nabla I(\mathbf{s}_i)|$  is the absolute value of image intensity gradient. In the proposed method,  $|\nabla I(\mathbf{s}_i)|$  is given by the following equations depending on what region the pixel belongs to.

$$|\nabla I(\mathbf{s}_i)| = \begin{cases} |I(u_i + 1, v_i) - I(u_i, v_i)| + |I(u_i, v_i + 1) - I(u_i, v_i)| & \text{if } (u, v) \notin R_3, R_4 \\ |I(u_i + 1, v_i) - I(u_i, v_i)| & \text{if } (u, v) \in R_3 \\ |I(u_i, v_i + 1) - I(u_i, v_i)| & \text{if } (u, v) \in R_4 \end{cases} \quad (11)$$

Equation (11) shows that the contour model should not be attracted to the edges belonging to the background grid lines. The horizontal and the vertical background grid lines are regarded as edges that have large intensity gradients along the vertical and the horizontal directions, respectively. Therefore, we make directionally selective calculation of intensity gradients for pixels belonging to region  $R_3$  or  $R_4$  in Eq.(11).

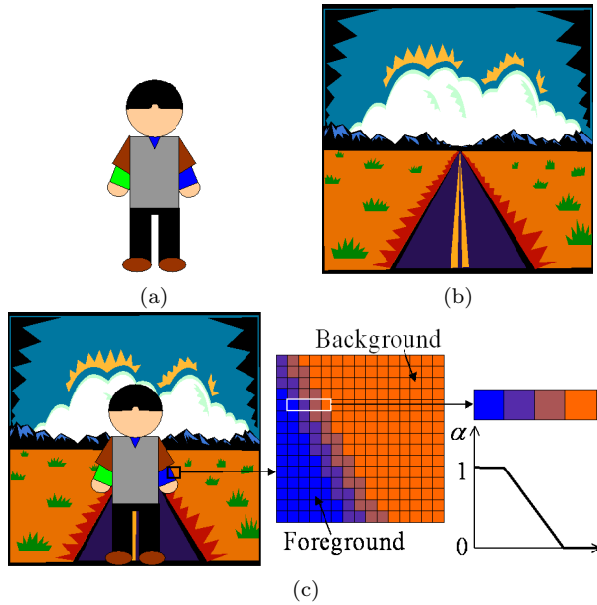
## 2.4 Image Composition

The extracted foreground image and another background image are combined by using Eq.(1) (Fig.4(f)). The alpha values for the background pixels are 0. The alpha values for the foreground pixels are determined by the distance of each foreground pixel from the background region. Let  $d$  be the distance from the background region, and  $d_0$  be the farthest distance beyond which the alpha value becomes constantly 1. Then we determine the alpha value of the pixels as  $d/d_0$ . Figure 13(a) and 13(b) show a foreground and background images, respectively, and Fig.13(c) shows the alpha value variation for foreground boundary pixels.

## 3. Experiments

Experiments were performed in an indoor environment. In the experiments, we selected blue and green as the





**Fig. 13** Image composition. (a) Foreground image. (b) Background image. (c) The alpha value variation.

colors of the checker pattern background, because they are complementary colors of human skin color and generally used in chroma key with a unicolor background. The pitch of the checker pattern was  $30\text{mm} \times 30\text{mm}$  to interpolate the boundary of  $R_6$  region precisely.

The size of the image was  $1600 \times 1200$  pixels. The threshold value  $T$  for a color extraction in Eq.(4) was 20. The length  $l$  of regions  $R_3$  and  $R_4$  was 4. The weighting factors  $w_{sp1}$ ,  $w_{sp2}$ ,  $w_{image}$ ,  $w_{area}$  for Snakes were 30, 3, 2, 1, respectively. In approximation of the background grid lines by a least mean square method, we used quartic equations to fit. Distance  $d_0$  for the alpha value determination was 2, where the distance was given in 8-neighborhood definition. All parameters were determined experimentally by manual search for optimal ones to give good results. These parameters were unchanged throughout the experiments shown in Figs.16, 17 and 18.

Figures 14, 15 and 16 show the results of region extraction using a unicolor, a stripe and a checker pattern background, respectively. Figures 14(a), 15(a) and 16(a) are the original images to segment, where sheets of the same paper used as the background are put on the foreground person in order to confirm the validity of the proposed method. Figure 14(b) shows that the foreground regions whose colors are the same as the background color can not be extracted. Figure 15(b) shows that if the foreground regions have the same colors as the background and have parallel contours with the background stripes, they can not be extracted. Figure 16(b) shows the result of background color extraction. Extracted background regions  $R_1$  and  $R_2$  are represented by white pixels in the figure. Figure 16(c) shows the result of detection of region  $R_7$ . Figure 16(d)

shows that the foreground regions whose colors are the same as the background color are extracted without fail. Figure 16(e) shows the result of the image composition of the extracted foreground and another background.

To confirm the validity of our method, the foreground region extracted by our method is compared with the result extracted manually. Figure 16(a) was used for validation. The number of foreground object pixels extracted manually was 340952, and our method extracted 338029 pixels as foreground object pixels. Our method failed to extract 3668 foreground pixels, and extracted 745 spurious pixels. These numbers show that errors of region extraction by our method are few.

Figure 17 shows other experimental results, where (a), (b), (c) show a foreground image, a background image, and the result of the image composition, respectively. The method has been verified in an indoor environment with a lot of people whose clothes were diverse in color. Figure 18 shows results for a moving image sequence, where (a), (b) show examples of foreground images and results of image composition, respectively.

#### 4. Conclusions

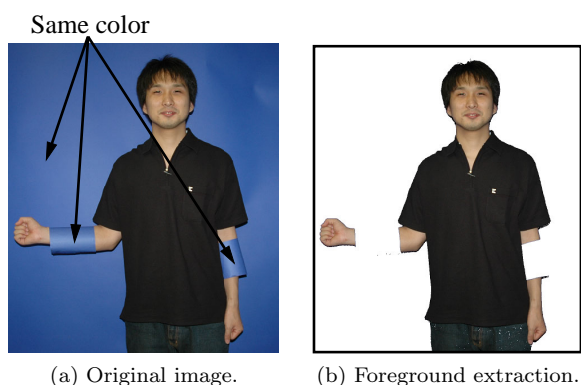
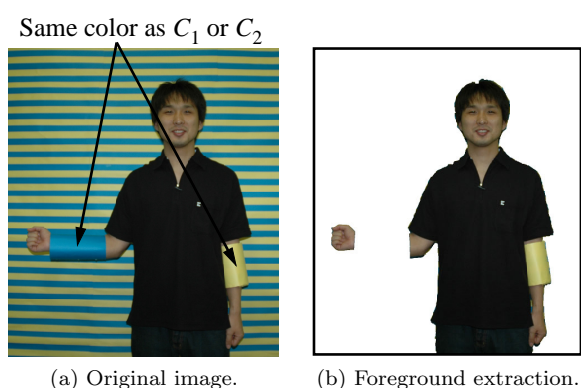
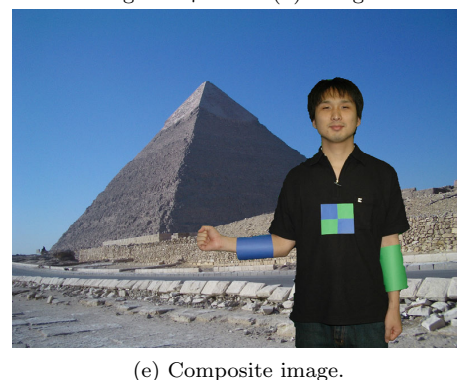
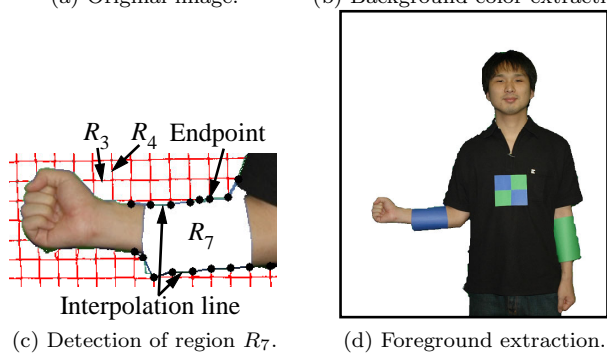
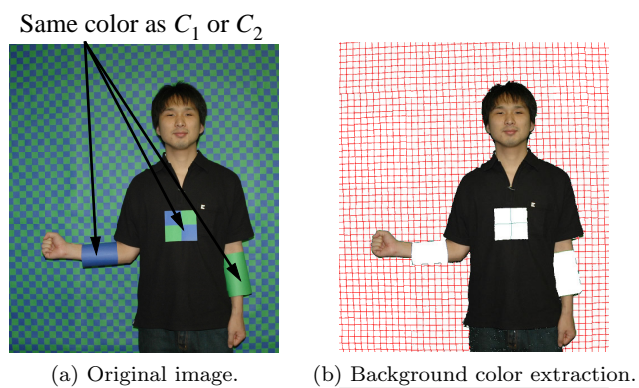
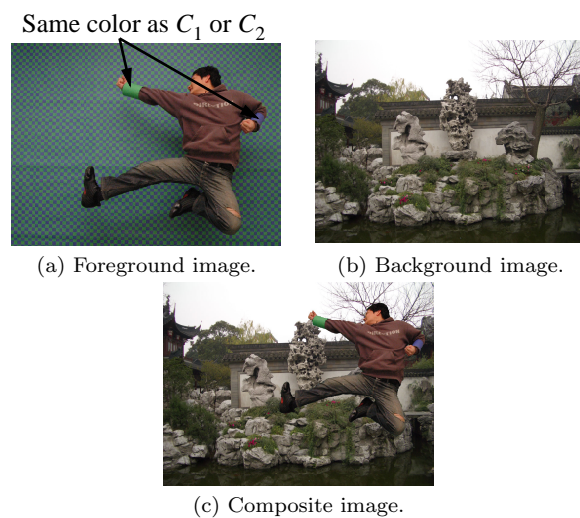
In this paper, we proposed a new region extraction method using chroma key with a two-tone checker pattern background. The method solves the problem in conventional chroma key techniques that foreground objects become transparent if their colors are the same as the background color. The method utilizes the adjacency condition between two-tone regions of the background and the geometrical information of the background grid lines. Experimental results show that the foreground objects can be segmented exactly from the background regardless of the colors of the foreground objects.

Although our proposed method can work successfully, we should make the following improvements.

1. The parameters for image processing should be determined automatically based on appropriate criteria.
2. The alpha value of each pixel should be estimated exactly by employing related methods, e.g. [16], [17].
3. When applying the method to a video sequence, we should take the advantage of interframe correlation. The parameters and the background grid-line geometry obtained in the first frame can be utilized in processing of succeeding frames so that the total processing time will be much shorter.

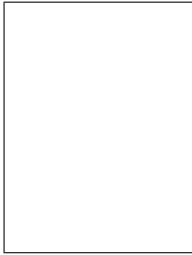
#### Acknowledgments

This research was partially supported by the Ministry of Education, Culture, Sports, Science and Technology, Grant-in-Aid for Scientific Research (C), 17500067.

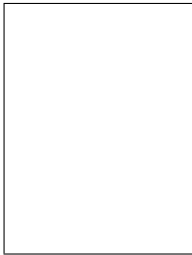
**Fig. 14** Result of region extraction using a unicolor background.**Fig. 15** Result of region extraction using a stripe background.**Fig. 18** Result for a moving image sequence.**Fig. 16** Result of region extraction using a checker pattern background. (In (b), background regions  $R_1$  and  $R_2$  are represented by white pixels.)**Fig. 17** Another example of region extraction using a checker pattern background.

## References

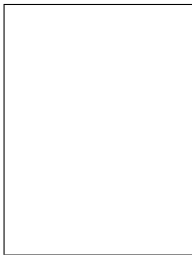
- [1] T. Porter and T. Duff, "Compositing digital images," *Computer Graphics (Proc. SIGGRAPH84)*, vol.18, no.3, pp.253–259, 1984.
- [2] S. Gibbs, C. Arapis, C. Breiteneder, V. Lalioti, S. Mostafawy, and J. Speier, "Virtual studios: An overview," *IEEE Multimedia*, vol.5, no.1, pp.18–35, 1998.
- [3] A. Wojdala, "Challenges of virtual set technology," *IEEE Multimedia*, vol.5, no.1, pp.50–57, 1998.
- [4] K.-S. Fu and J. K. Mui, "A survey on image segmentation," *Pattern Recognition*, vol.13, pp.3–16, 1981.
- [5] W. Skarbek and A. Koschan, "Colour image segmentation - A survey," *Technical Report 94-32*, Technical University of Berlin, Department of Computer Science, 1994.
- [6] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int. J. Comput. Vis.*, vol.1, no.4, pp.321–331, 1988.
- [7] T. Mitsunaga, Y. Yokoyama, and T. Totsuka, "AutoKey: Human assisted key extraction," *Computer Graphics (Proc. SIGGRAPH95)*, pp.265–272, 1995.
- [8] R. J. Qian and M. I. Sezan, "Video background replacement without a blue screen," *Proc. IEEE International Conference on Image Processing*, pp.143–146, 1999.
- [9] S. Shimoda, M. Hayashi, and Y. Kanatsugu, "New chroma-key imaging technique with hi-vision background," *IEEE Trans. Broadcast.*, vol.35, no.4, pp.357–361, 1989.
- [10] T. Kanade, A. Yoshida, K. Oda, H. Kano, and M. Tanaka, "A stereo machine for video-rate dense depth mapping and its new applications," *Proc. 1996 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.196–202, 1996.
- [11] M. Kawakita, K. Iizuka, T. Aida, H. Kikuchi, H. Fujikake, J. Yonai, and K. Takizawa, "Axi-vision camera (real-time distance-mapping camera)," *Appl. Opt.*, vol.39, no.22, pp.3931–3939, 2000.
- [12] Y. Mishima, "A software chromakeyer using polyhedric slice," *Proc. NICOGRAPH92*, pp.44–52, Nov. 1992.
- [13] D. E. Zongker, D. M. Werner, B. Curless, and D. H. Salesin, "Environment matting and compositing," *Computer Graphics (Proc. SIGGRAPH99)*, pp.205–214, 1999.
- [14] M. A. Ruzon and C. Tomasi, "Alpha estimation in natural images," *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.18–25, 2000.
- [15] P. Hillman, J. Hannah, and D. Renshaw, "Alpha channel estimation in high resolution images and image sequences," *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol.1, pp.1063–1068, 2001.
- [16] Y.-Y. Chuang, B. Curless, D. H. Salesin, and R. Szeliski, "A Bayesian approach to digital matting," *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol.2, pp.264–271, 2001.
- [17] J. Sun, J. Jia, C.-K. Tang, and H.-Y. Shum, "Poisson matting," *Computer Graphics (Proc. SIGGRAPH2004)*, pp.315–321, 2004.
- [18] A. R. Smith and J. F. Blinn, "Blue screen matting," *Computer Graphics (Proc. SIGGRAPH96)*, pp.259–268, 1996.
- [19] A. Yamashita, T. Kaneko, S. Matsushita, and K. T. Miura, "Region extraction with chromakey using stripe backgrounds," *IEICE Trans. Inf. & Syst.*, vol.E87-D, no.1, pp.66–73, Jan. 2004.
- [20] S. Araki, N. Yokoya, H. Iwasa, and H. Takemura, "A new splitting active contour model based on crossing detection," *Proc. 2nd Asian Conf. on Computer Vision (ACCV'95)*, vol.II, pp.346–350, 1995.



**Hiroki Agata** received B.E. from the Department of Mechanical Engineering, Shizuoka University, in 2005. He has been a master student in the Department of Mechanical Engineering, Shizuoka University, since 2005.



**Atsushi Yamashita** received B.E., M.E., and Ph.D. from the Department of Precision Engineering, the University of Tokyo, in 1996, 1998 and 2001, respectively. From 1998 to 2001, he was a Junior Research Associate in the RIKEN (Institute of Physical and Chemical Research). He has been a Assistant Professor in the Department of Mechanical Engineering, Shizuoka University, since 2001. He has been a Visiting Associate in the Department of Mechanical Engineering, California Institute of Technology, since 2006. His research interests are image processing, computer vision, robot motion planning, and mobile robot mechanism. He is a member of IEEE, IPSJ, IEEJ, SICE, ITE, RSJ, JSPE, and JSME. E-mail: yamashita@ieee.org



**Toru Kaneko** received B.E. and M.E. from the Department of Applied Physics, the University of Tokyo, in 1972 and 1974, respectively. From 1974 to 1997, he joined Nippon Telegraph and Telephone Corporation (NTT). He received Ph.D. from the University of Tokyo in 1986. He has been a Professor in the Department of Mechanical Engineering, Shizuoka University, since 1997. His research interests are computer vision and image processing.

He is a member of IEEE, IPSJ, ITE, RSJ, etc.