

Region Extraction Using Chroma Key with a Checker Pattern Background

Hiroki AGATA, Atsushi YAMASHITA and Toru KANEKO
Faculty of Engineering, Shizuoka University
3-5-1 Johoku, Hamamatsu-shi, Shizuoka 432-8561, Japan

Abstract—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 they have the same color with the background. The method utilizes the adjacency condition between two-tone regions of the background and the geometrical information of the background grid line. Experimental results show that the foreground objects can be segmented exactly from the background regardless of the colors of the foreground objects.

I. 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].

To perform image composition, objects of interest must be segmented from images, and there are many studies about image segmentation [3], [4], e.g. pixel-based, area-based, edge-based, and physics-based ones. For example, Snakes [5] 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 [6]. 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 [7]. 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 [8]. This method is a fundamental technique of virtual studios [2].

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 [9]. 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 [10]. 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 it.

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 behind it. 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 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 [11]-[16].

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(a)). To solve this problem, Smith and Blinn proposed a blue screen matting method that allows foreground objects to be shot against two backing colors [17]. This method can extract the foreground region whose colors are same with the background color. However, 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 propose a new method that makes it possible to extract objects from a background precisely even if objects have a color similar to the background. The method uses a two-tone checker pattern background so that it can extract foreground objects by analyzing geometrical conditions even if they have the same color as the background. (Fig. 1(b))

II. CHROMA KEY WITH A CHECKER PATTERN BACKGROUND

The procedure of the method consists of four steps; background color extraction (Fig. 2(a)), background grid line extraction (Fig. 2(b)(c)), foreground extraction (Fig. 2(d)(e)), and image composition (Fig. 2(f)).

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

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

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. 2(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:

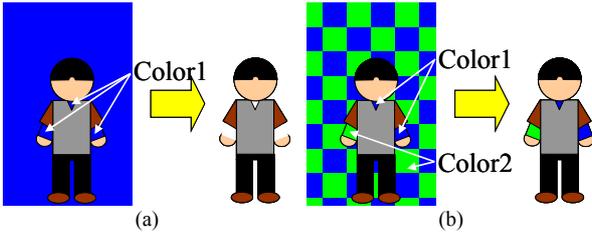


Figure 1. Region extraction with chroma key. (a) Conventional method. (b) The proposed method.

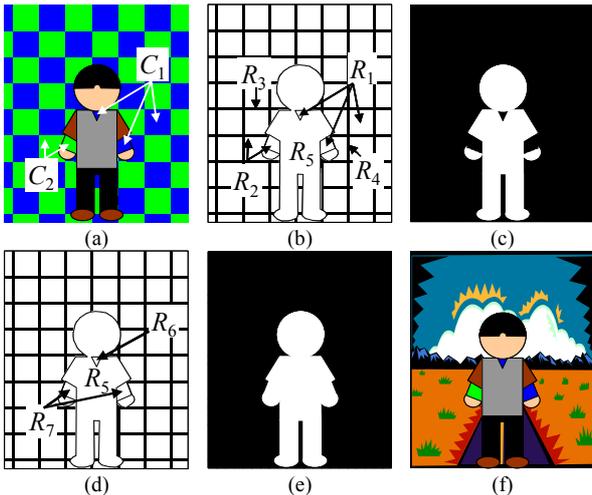


Figure 2. Region extraction. (a) Original image. (b) Region segmentation without the information of checker pattern. (c) Foreground extraction by the result of (b). (d) Region segmentation using the information of checker pattern. (e) Foreground extraction by the result of (d). (f) Image composition.

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

Fig. 3 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. 4(a). The colors in these reference areas are divided into C_1, C_2 and C_3 in the HLS color space by using K-mean clustering. Fig. 4(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. As a reason to use the HLS color space, 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:

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

B. 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

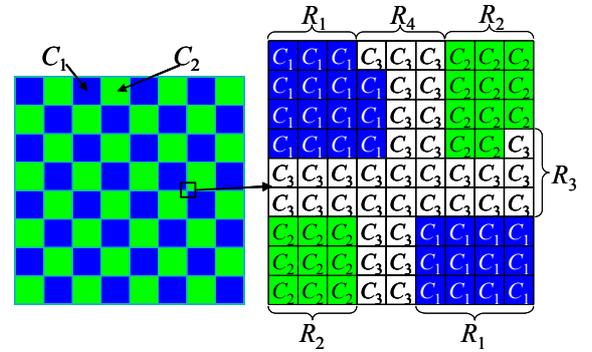


Figure 3. Background regions and their colors.

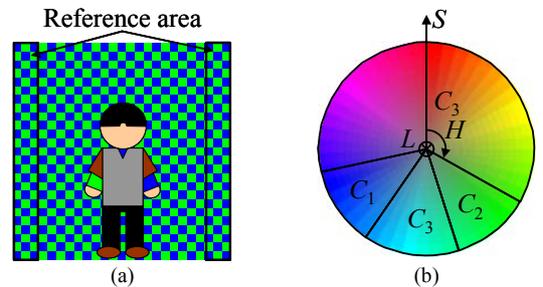


Figure 4. Background color estimation. (a) Reference areas. (b) HLS color space.

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:

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

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

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. 5, 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. 6, 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 line exists at the top of the image as shown in Fig. 7(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.

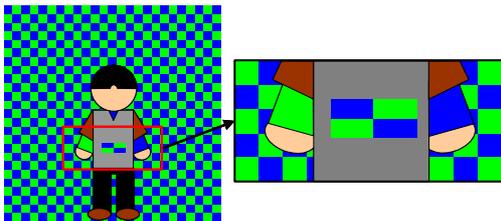


Figure 5. Checker pattern in the foreground.

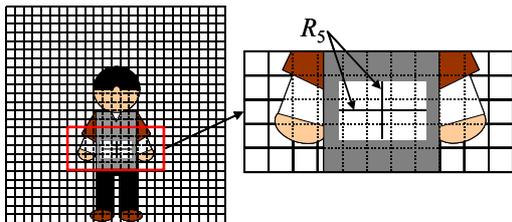


Figure 6. Background grid line approximation.

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. 7(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 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. 2(c). Those are regions which belong to the foreground object but have the same color as the background. As shown in Fig. 2(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.

C. 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. 8, top).

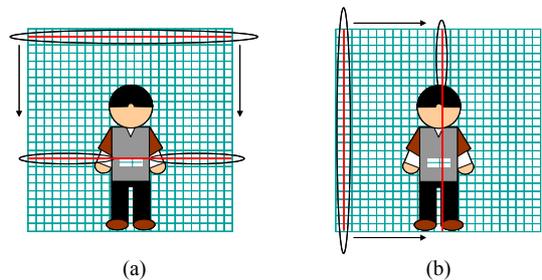


Figure 7. Background grid line estimation. (a) Horizontal line. (b) Vertical line.

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. 8, 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. 9). The region containing the back-ground grid line is regarded as the background, and the other is regarded as the foreground region R_7 .

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. 2(d), (e)). However, 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 line connection. Therefore, we need to execute post processing for contour refinement which is realized by Snakes [5].

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 [18]. These energies are defined as:

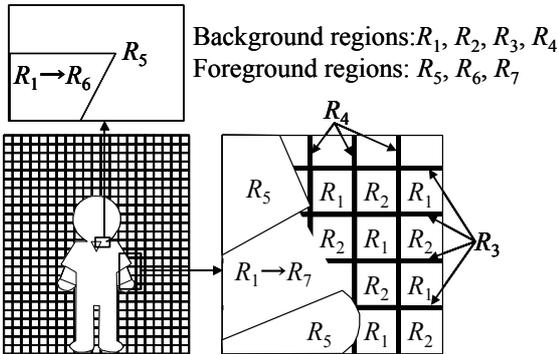


Figure 8. Foreground region whose color is same as the background. Top: Inside the foreground (R_1 is reclassified to R_6). Right: Neighboring to the background (R_1 is reclassified to R_7).

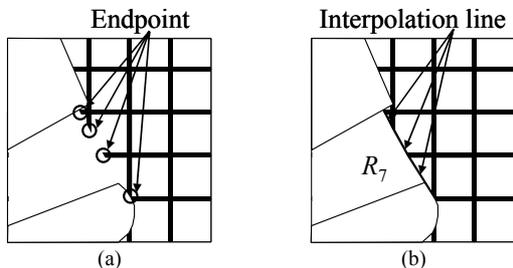


Figure 9. Determination of region R_7 . (a) Grid line endpoints. (b) Interpolation lines.

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

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

$$E_{area}(\mathbf{s}_i) = \sum_{i=1}^n w_{area} \{u_i (v_{i+1} - v_i) - (u_{i+1} - u_i) v_i\}, \quad (10)$$

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.

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

In our method, background grid lines behave as edges to which the contour model is attracted, if image intensity gradient is calculated in each direction. Therefore, we calculate image intensity gradient with one direction at background grid lines.

D. Image Composition

The extracted foreground image and another back-ground image are combined by using Eq. (1) (Fig. 2(f)). The alpha values for the foreground and the background are 1 and 0, respectively. The alpha values around the boundaries between the foreground and the background vary in the range 0-1 according to geometrical and optical conditions. In this paper, the alpha values vary in the range 0-1 linearly from the background to the foreground (Fig. 10).

III. EXPERIMENTS

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

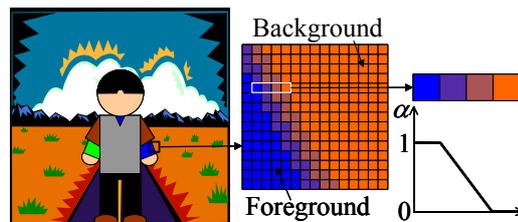


Figure 10. The alpha value variation.

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 1600×1200 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. All parameters were determined experimentally by manual search for optimal ones to give good results. These parameters were unchanged throughout the experiments shown in Fig. 12, 13, 14.

Fig. 11 and 12 show the results of region extraction using a unicolor and a checker pattern background, respectively. Fig. 11(a) and 12(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. Fig. 11(b) shows that the foreground regions whose colors are same with the background can not be extracted. Fig. 12(b) shows the result of background color extraction. Extracted background regions R_1 and R_2 are represented by white pixels in the figure. Fig. 12(c) shows the result of detection of region R_7 . Fig. 12(d) shows that the foreground regions whose colors are same with the background are extracted without fail. Fig. 12(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. Fig. 12(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.

Fig. 13 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. Fig. 14 shows results for a moving image sequence, where (a), (b) show examples of foreground images and results of image composition, respectively.

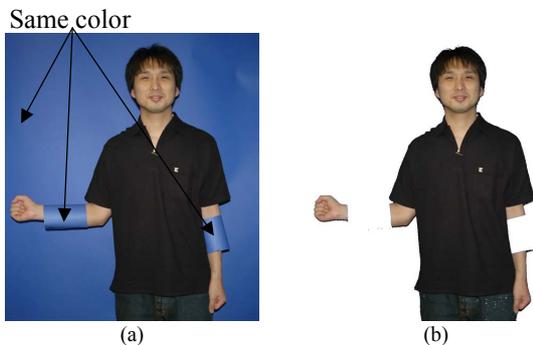


Figure 11. Result of region extraction using a unicolor background. (a) Original image. (b) Foreground extraction.

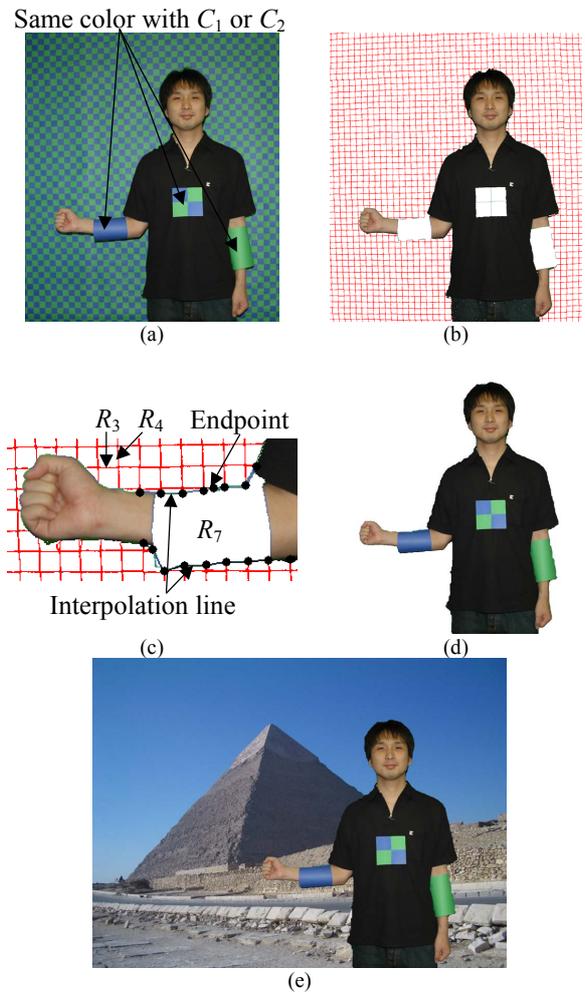


Figure 12. Result of region extraction using a checker pattern background. (a) Original image. (b) Background color extraction (Background regions R_1 and R_2 are represented by white pixels). (c) Detection of region R_7 . (d) Foreground extraction. (e) Image composition.

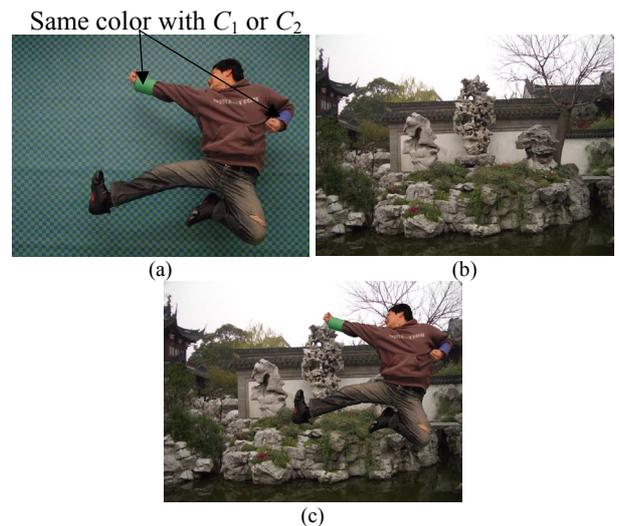


Figure 13. Another example of region extraction using a checker pattern background. (a) Foreground image. (b) Background image. (c) Image composition.

Same color with C_1 or C_2



Figure 14. Result for a moving image sequence. (a) Original image. (b) Image composition.

IV. 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 they have the same color with the background. 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. [15], [16].

ACKNOWLEDGMENTS

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

REFERENCES

- [1] T. Porter and T. Duff, "Compositing Digital Images", *Computer Graphics (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] K.-S. Fu and J. K. Mui, "A Survey on Image Segmentation", *Pattern Recognition*, Vol.13, pp.3-16, 1981.
- [4] W. Skarbek and A. Koschan, "Colour Image Segmentation - A Survey", *Technical Report 94-32, Technical University of Berlin, Department of Computer Science*, 1994.
- [5] M. Kass, A. Witkin and D. Terzopoulos, "Snakes: Active Contour Models", *International Journal of Computer Vision*, Vol.1, No.4, pp.321-331, 1988.
- [6] T. Mitsunaga, Y. Yokoyama and T. Totsuka, "AutoKey: Human Assisted Key Extraction", *Computer Graphics (Proceedings of SIGGRAPH95)*, pp.265-272, 1995.
- [7] R. J. Qian and M. I. Sezan, "Video Background Replacement without A Blue Screen", *Proceedings of IEEE International Conference on Image Processing*, pp.143-146, 1999.
- [8] S. Shimoda, M. Hayashi and Y. Kanatsugu, "New Chroma-key Imaging Technique with Hi-Vision Background", *IEEE Transactions on Broadcasting*, Vol.35, No.4, pp.357-361, 1989.
- [9] T. Kanade, A. Yoshida, K. Oda, H. Kano and M. Tanaka, "A Stereo Machine for Video-Rate Dense Depth Mapping and its New Applications", *Proceedings of the 1996 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.196-202, 1996.
- [10] M. Kawakita, K. Iizuka, T. Aida, H. Kikuchi, H. Fujikake, J. Yonai and K. Takizawa, "Axi-Vision Camera (real-time distance-mapping camera)", *Applied Optics*, Vol.39, No.22, pp.3931-3939, 2000.
- [11] Y. Mishima, "A Software Chromakeyer Using Polyhedral Slice", *Proceedings of NICOGRAPH92*, pp.44-52, November 1992.
- [12] D. E. Zongker, D. M. Werner, B. Curless and D. H. Salesin, "Environment Matting and Compositing", *Computer Graphics (Proceedings of SIGGRAPH99)*, pp.205-214, 1999.
- [13] M. A. Ruzon and C. Tomasi, "Alpha Estimation in Natural Images", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.18-25, 2000.
- [14] P. Hillman, J. Hannah and D. Renshaw, "Alpha Channel Estimation in High Resolution Images and Image Sequences", *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol.1, pp.1063-1068, 2001.
- [15] Y.-Y. Chuang, B. Curless, D. H. Salesin and R. Szeliski, "A Bayesian Approach to Digital Matting", *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol.2, pp.264-271, 2001.
- [16] J. Sun, J. Jia, C.-K. Tang and H.-Y. Shum, "Poisson Matting", *Computer Graphics (Proceedings of SIGGRAPH 2004)*, pp.315-321, 2004.
- [17] A. R. Smith and J. F. Blinn, "Blue Screen Matting", *Computer Graphics (Proceedings of SIGGRAPH96)*, pp.259-268, 1996.
- [18] S. Araki, N. Yokoya, H. Iwasa and H. Takemura, "A New Splitting Active Contour Model Based on Crossing Detection", *Proceedings of 2nd Asian Conf. on Computer Vision (ACCV'95)*, Vol. II, pp.346-350, 1995.