

Separation of Reflection and Transparency Based on Spatiotemporal Analysis for Outdoor Scene

THANDA OO,[†] HIROSHI KAWASAKI,[†] YUTAKA OHSAWA[†]
and KATSUSHI IKEUCHI^{††}

The effect of reflection and transparency, which results from shiny or glass-like transparent materials, is superimposed on captured images of many actual outdoor scenes. The presence of such an incidental effect in a captured image has made it difficult to apply computer vision algorithms and has led to erroneous results. Moreover, it disturbs the texture acquisition of an outdoor scene, an important topic for the Computer Graphics (CG) and Intelligence Transportation System (ITS) community. This paper presents an optimal method for the automatic separation of reflected and transparent layers even if the scene is complicated with view-dependent effects and depth disparity in a 3D environment. The method is based on epipolar plane image (EPI) analysis. The method is not like the conventional edge-based EPI analysis, but instead it is a color-based analysis. First, we separate EPI into two layers by our original color-based EPI analysis. Then, original image sequence is separated into reflected and transparent layers by using the separated EPI. To demonstrate the effectiveness of our method, we implement the algorithm and present the results of experiments using synthesized and real scene images including indoor and outdoor scenes.

1. Introduction

The presence of reflection and transparency in the same image is caused by shiny or glass-like transparent materials and often occurs in many natural scenes. The observed color of such a scene is a combination of the light transmitted from an actual object behind the glass and a virtual object on the glass, which occurs when objects exist in front of the glass. This combination causes errors in some computer vision algorithms, such as the algorithms for object tracking. In addition, this effect strongly disturbs the texture acquisition of a real-world scene, which is a critical research area in computer vision and computer graphics, including 3D urban modeling projects. One possible solution to this problem is to separate the component images. Many researchers have tried to separate the reflection and transparent components, and many valuable methods have already been proposed. However, most proposals have not considered outdoor scenes, which usually consist of 3D objects in large depth disparity.

In this paper, we propose a method to acquire texture without a reflection component from an outdoor image sequence for the purpose of urban modeling. To achieve this, we

propose a new method to separate the reflection and transparency layers that have been caused by objects that exist in front of glass and behind glass, even if the objects have large depth disparity and a camera moves drastically. Our method is based on epipolar plane image (EPI) analysis. Unlike conventional EPI analysis, which usually analyzes edges, we propose a color-based analysis of EPI, which can robustly separate the reflection and transparency components.

The remainder of this paper is organized as follows. In Section 2, we discuss related works. A brief explanation of EPI-based edge and color analysis is included in Section 3. Section 4 introduces a system outline and a new method to separate the reflection and transparency components that we have proposed. Experiments and results can be seen in Section 5. Finally, in Section 6, we offer our conclusion.

2. Related Works

There has been much research conducted to separate reflection and transparency component layers in various ways. A number of studies based on independent component analysis (ICA)^{2)~4)} have been conducted. A method has been proposed for separating reflection from images by applying an analytic version of ICA³⁾, and, based on the same principle, dividing the whole image into partial images⁴⁾. Another approach is to process block images and

[†] Department of Computer Sciences and Engineering, Saitama University

^{††} Institute of Industrial Science, The University of Tokyo

apply sparse ICA to separate reflection component²⁾.

Other methods using an optical property to separate actual and virtual objects have been proposed. Schechner et al. investigated a method that utilizes polarization^{10)~12)}. However, these approaches need a polarization filter to be used with the camera, and thus, it cannot be applied to a moving camera. An effort was also made to separate transparent layers using focus^{8),9)}. This separation method focused on a single layer and used blur kernels to enhance it⁹⁾. The method also proposed a self-calibration method of blur kernels⁸⁾.

Layer-based methods for the separation of superimposed layers have been widely proposed^{5),7),14)}. These methods assume that the sampled image is a linear combination of several layers, typically two layers. A method has been proposed to recover the transparent layer by fusing the same regions that are segmented by motion analysis from successive frames and averaging the dominant object⁵⁾. Sarel et al. have proposed transparent layer separation through layer information exchange⁷⁾, which minimizes the structural correlation across two images. In Irani, et al.'s work^{5),7)}, input image sequences are captured by a static camera, so that the target scene is static for all image frames. This method assumes a static camera and cannot be applied to a moving camera.

Layer extraction from image sequence by using the constrained least squares method has been proposed¹⁴⁾. These researchers applied their method to a moving camera and could successfully extract the component layers from an image sequence. However, they only performed their experiment under controlled experimental situations such as restricting the shape of the objects to 2D planes such as a framed picture, and thus, their method did not assume depth disparity and would fail for an actual 3D environment.

In an outdoor scene, all objects have different depths, and these depth disparities lead to different motions within a layer when a camera moves. As a result, all of the above-mentioned methods cannot be applied to an actual outdoor environment. With our proposed method, 3D objects in both layers could be treated properly by analyzing EPI, and input images can be separated into two layers.

3. EPI Analysis

3.1 Definition of an EPI

In this section, we explain how we create an epipolar plane image and how we perform an EPI analysis based on the nature and geometrical appearance of EPI. The first step is to make a spatiotemporal image volume and slice it horizontally to acquire the EPI. The EPI volume can be constructed by taking a series of images with a moving camera and accumulating these images along the temporal direction, as shown in **Fig. 1**. The camera's motion is assumed to be a constant speed and a straight path, so that the analysis becomes easy and robust. Such assumption is impractical for actual data acquisition of an outdoor scene. Therefore, in this paper, we will eliminate these assumptions for greater convenience. Details are described in Section 4.1.

On the EPI, the frontal surface of any object appears as an area bounded by two distinct parallel boundaries. In the rest of this paper, we call this area the EPI-strip, or strip. Although we restrict the camera motion along a straight line, the depth d of all objects are not the same in the real world; therefore, all EPI-strips do not lie in a parallel direction. This depth difference gives special characteristics to the EPI, as shown in Fig. 1 (bottom). A detailed explanation is included in the next section.

3.2 Edge-based EPI Analysis

If there is no transparent object in the scene, as in **Fig. 2** (left), we can produce an EPI as shown in Fig. 2 (right). While object 2 can be captured for all frames, objects 1 and 3 appear only for a short period in the camera view. The inclination angle θ of an EPI-strip can be describe as

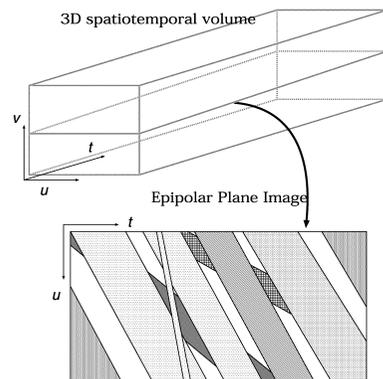


Fig. 1 An EPI.

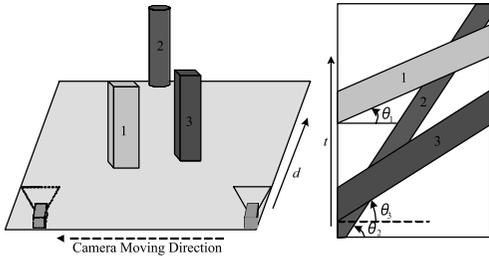


Fig. 2 Appearance and nature of actual objects in the EPI.

$$\tan\theta = \frac{d}{vf} \quad (1)$$

therefore,

$$\theta \propto d \quad (2)$$

where v is camera velocity and f represents a focal length. θ_1 , θ_2 , and θ_3 in Fig.2 represent the inclination angles of EPI-strips 1, 2, and 3, respectively, and we can clearly see that the inclination angles of the EPI-strips are directly proportional to the depth of the objects ($\theta_1 < \theta_3 < \theta_2$ with respect to $d_1 < d_3 < d_2$). Furthermore, strip 2 is totally covered by the other two opaque strips at the overlap areas, so we cannot see the object if there is an object in front of it. Therefore, the boundary edge of strip 2 cannot be connected at the overlap area and strip 2 is divided into three separate areas. Even though these three areas are separated, we can still understand that these three areas produce a single EPI-strip by analyzing the edge parallelism and color similarity^{1),15)}. With such an edge-based analysis, we can retrieve the 3D information from the EPI and the scene.

3.3 Color-based EPI Analysis

To conduct a conventional EPI analysis, we usually assume that the object's appearance does not change drastically; that is, the object's appearance does not depend on the view direction, and there is no reflection or transparency. However, in reality, the color sometimes drastically changes depending on the view direction because of the superimposed reflection and transparency, and other effects such as specularly¹³⁾. To conduct a further analysis with an EPI in a real-world scene, we have to understand how color is produced on an EPI and we must include the effects of reflection and transparency. A color change on an EPI can be basically explained by three reasons: first, changes of material or color of the target objects; second, the effect of a cast shadow; and third, the effect of reflection. It should be noted that we

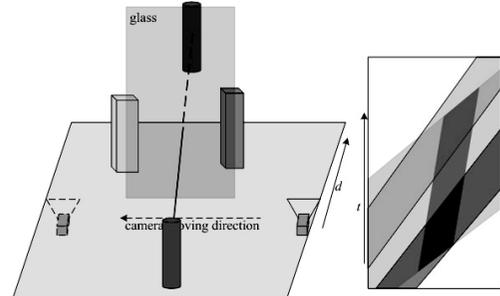


Fig. 3 Appearance and nature of transparent and reflected objects in an EPI.

do not consider complicated bi-directional reflectance distribution function (BRDF) in this paper. Most colors on an EPI can be explained by the combination of those three reasons.

The second reason, the color change occurred between a cast shadow and no cast shadow area on an EPI, can be considered to be of the same nature as the first reason. This is because a cast shadow does not change its appearance depending on view direction, and thus, the EPI-strip produced by a cast shadow on the object can be treated as the EPI-strip of the actual object. Based on this, it can be basically considered that a cast shadow has no influence on the separation of reflection and transparency except that the cast shadow area is so dark that the pixel value is beneath the dynamic range of a sensor.

With regard to third reason, if the target object is opaque, the effect of reflection can be seen as specularly, whereas, if the target object is transparent, the effect of reflection can be seen as superimposed layers on the object. The motion and appearance of specularities in an EPI are already analyzed¹³⁾. Therefore, in this paper we analyze the characteristics of reflection and transparency on an EPI. As shown in **Fig. 3**, a virtual object is observed as if the actual object exists in front of glass; therefore, the object simply describes an EPI-strip on an EPI. However, since glass is transparent, the observed color is a mixture of actual and virtual objects. Therefore, whereas the virtual object makes a single band, its color changes abruptly when it intersects the EPI-strips of the actual objects and vice versa. Note that, under such conditions, we can still distinguish each EPI-strip robustly by a simple edge detection algorithm. Such distinction is usually difficult to achieve by simple image processing techniques, such as a motion tracking technique applied on

the original image sequence.

4. System Outline

The system comprises a moving camera, which is controlled by a motorized stage for an indoor experiment or mounted on the roof of the data acquisition car for an outdoor experiment. A preprocessing module recognizes the opaque areas in the captured image sequences, and estimates the path and speed difference between successive image frames to eliminate restrictions of EPI, such as a constant speed and a straight path. This module converts the image sequences as they are captured by constant speed and straight path, and constructs the EPIs. After that, the strips are defined on an EPI, and the layer separation of the EPI is performed by our proposed separation method. Finally, the separation of the original image is achieved by two ways called iterative and color clustering methods. **Figure 4** shows the flow diagram of the system outline.

4.1 Preprocessing of Captured Data

Most of EPI analysis research has restricted the camera motion to a constant speed and movement along a straight path. However, these restrictions can only be obeyed when we use synthesized input images or capture image sequences in an indoor environment. When the application area is moving to an outdoor environment, the above restrictions are no longer attainable and we need to extend our method free from these restrictions. In this section, we explain some possible methods and how we eliminate these restrictions.

The simplest way is to use additional devices such as GPS, vehicle speed sensors, gyro sensor, or some other special devices to compensate for the motion. Another idea is, if we divide the whole path into a number of short paths, these paths can be assumed to be straight and the speed is constant if the car does not run along an intentionally zigzag or sinuous path. With our proposed method, only small numbers of image frames are required to produce an EPI, because color-based EPI analysis is applied to the actual object (building), and the target building appears on a camera view only about 50 frames when car speed is 60 Km/hr. In this paper, we implemented and tested both methods.

Even if we applied color-based EPI analysis for almost straight path, a high-frequency vibration may still exist in the actual situation.

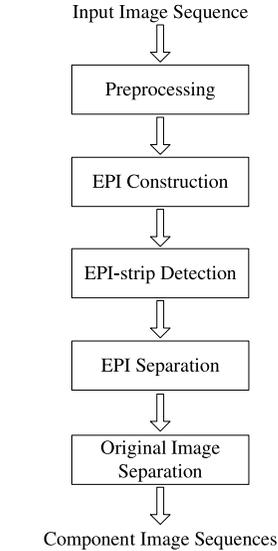


Fig. 4 System flow diagram.

To eliminate this hindrance to constant speed and straight path, a block matching algorithm is provided. Since a block matching algorithm requires a selection of a sample blocks from an opaque area, the recognition of the opaque area from an input image sequence is necessary. In this paper, we assume Lambertian for all objects; therefore, we could take a simple approach as follows. Because of the Lambertian assumption, the color of all pixels within a straight line on the EPI that is parallel to the EPI-strip's boundary is almost constant. Therefore, by investigating the color variation along those straight lines in the EPI, we could recognize the opaque area in the original input image sequence. **Figure 5** (b) shows the color variation map of the target building in Fig. 5 (a) for the R channel. The other two channels give similar results. The recognition can be done by thresholding all three channel results, and the thresholding result can be seen in Fig. 5 (c). For general purposes, further research is necessary.

After recognition, a sample block S is selected from the opaque area of the first frame of the original image sequence. The average pixel displacement between successive frames of the original image sequence can be estimated from the EPI and the expected match block E_k in the next frame can be calculated. Then the search area E to apply the block matching algorithm is predefined by expanding E_k with 10 pixels to all boundaries. The similarity between blocks can be evaluated on the basis of several

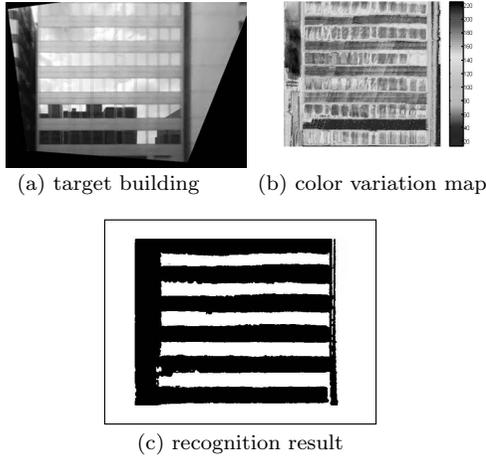


Fig. 5 Recognition of an opaque area.

methods, from which we adopted the normalized cross-correlation function:

$$\begin{aligned}
 NCC &= \frac{\sum_{x,y} (S_{(x,y)} - \bar{S})(E_{k(x,y)} - \bar{E}_k)}{\sqrt{(\sum_{x,y} (S_{(x,y)} - \bar{S})^2)(\sum_{x,y} (E_{k(x,y)} - \bar{E}_k)^2)}} \\
 &= \frac{\sum_{x,y} (S_{(x,y)} - \bar{S})(E_{k(x,y)} - \bar{E}_k)}{\sqrt{(\sum_{x,y} (S_{(x,y)} - \bar{S})^2)(\sum_{x,y} (E_{k(x,y)} - \bar{E}_k)^2)}} \quad (3)
 \end{aligned}$$

where, $\bar{S} = \text{mean}(S)$, $\bar{E}_k = \text{mean}(E_k)$, $E_k \subset E$ and (x, y) represents the image coordinate.

The block matching algorithm is applied by shifting pixel by pixel to find the maximum correlation block within the predefined matching area. The system iterates the matching algorithm to all image frames and produces a 2D matching array $(X_{diff}(i), Y_{diff}(i))$ $\{i = 2, 3, \dots, n|n = \text{number of image frames}\}$. In this paper, we assume that the low frequency effects can be adjusted by GPS and other devices. Therefore, X_{diff} can be considered as the speed difference between each successive frame, and Y_{diff} represents the vibrations and small waves on the road. Therefore Y_{diff} and X_{diff} can be used for adjusting both path and speed.

4.2 Defining Strips on an EPI

Since the camera is assumed to move linearly along a straight line by applying the preprocessing method, each object in the scene is bounded by two parallel lines on the EPI. As a result, parallel line detection by using Hough transform is sufficient to detect the boundary lines of the EPI-strips as shown in **Fig. 6** (c). We used only high energy peaks of the Hough space as shown in Fig. 6 (a) to detect the distinct edges

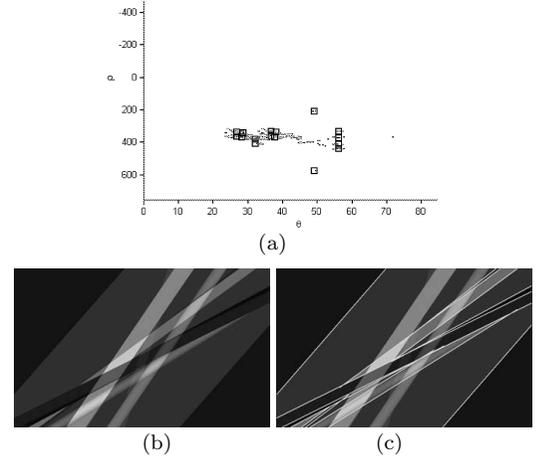


Fig. 6 (a) Hough transform image of (b), white rectangles represent for maximum peaks. (b) The synthesize EPI. (c) The detected boundary lines ($\theta < 50$).

such as the boundary of the objects. The inclination angle of the boundary line in the EPI is directly proportional to the depth of the object.

In a city, generally the virtual object exists farther away from the camera than the actual objects inside the building. In this case, the inclination angles of the EPI-strips produced by the virtual objects are steeper than those of actual objects. Therefore, we can easily recognize which layer the EPI-strip belongs to by simply dividing EPI-strips at the threshold angle between the virtual and the actual object. This recognition is necessary for the separation method, which will be described in the next section.

4.3 Separation of the EPI

We now describe a technique to separate the two component layers of the EPI and estimate the underlying original colors of the overlap regions in the EPI. Considering the presence of both the reflection and transparency components at the same image point, if we suppose the color of the overlap area is the linear combination of two color components, the observed color of that image point can be described as

$$M_c(x, y) = f_t \sigma_c^{act}(x, y) + f_r \sigma_c^{virt}(x, y) \quad (4)$$

where c represents the type of sensor (r, g and b), (x, y) is the two-dimensional image coordinate, $\sigma_c^{act}(x, y)$, $\sigma_c^{virt}(x, y)$ represent the colors of the actual and virtual objects, respectively, and f_t and f_r are the factors of transparency and reflection, respectively. In this paper, we assume the factor of reflection and transparency

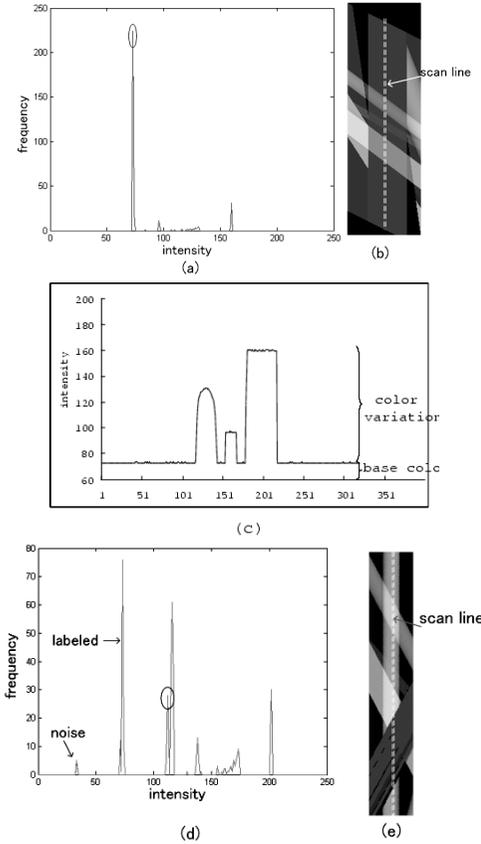


Fig. 7 Example histograms of scanned lines; black circles show the thresholding results. (a) shows the thresholding result of a scanned line on (b). (c) shows color intensity along a scan line on (b). (d) shows the thresholding result of a scanned line on (e).

is constant for all image frames. For simplicity, Eq. (4) can be rewritten as

$$M_c(x, y) = \hat{\sigma}_c^{act}(x, y) + \hat{\sigma}_c^{virt}(x, y). \quad (5)$$

In other words, every pixel value in the EPI is a sum of two colors, and each pixel in the EPI necessarily belongs to both EPI-strips of the reflected and transparent layers. Let us consider the straight line in the EPI-strip of the transparent layer, which is parallel to the EPI-strip’s boundary as the scan line. Note that the selection of the transparent layer only is assured in our experiment as explained in the previous section. Since the scan line represents for the same image point on the actual object, the color value along the line should be nearly constant. Therefore, the color variation along the straight scan line can be considered as the color variation of the other (reflected) layer. Thus, we can consider the minimum value along the scan line

as the color value of actual object as shown in **Fig. 7 (c)**.

Certainly, this is not the actual color for the actual object, and it remains the base color ambiguity. For this reason, if the reflected image in all frames has no texture, the method will do nothing because the color of all the pixels along a scan line in the EPI is nearly constant. Although such ambiguity remains, the result can be effectively applied for most computer vision algorithms. Furthermore, for texture acquisition purposes, human interaction can produce a reasonable result.

In our actual implementation, first we rectify (shear) the EPI along the inclination angle of the EPI-strip, so that the scan line becomes vertical, making it easy to scan. Then, we estimate the minimum color value along the scan line and assume it as the initial color value for that scan line. At the same time, we label the whole pixels along the scan line in order to avoid multiple scanning of the same pixel. The algorithm starts from the smallest angle of Hough transformation result to the largest angle until all pixels are labeled. After completion, another layer can be acquired by simply subtracting the estimated EPI layer from the original EPI .

With regard to minimum value estimation, there is a noise observed on the EPI produced from actually captured video data. Therefore, the histogram thresholding method is necessary for real-EPI separation. Our method simply takes the pixel value of the first peak nearest to zero intensity which is larger than threshold.

An example of histogram thresholding methods can be seen in Fig. 7. Figure 7(a) shows the histogram thresholding result of a scan line from Fig. 7(b); in this case, there is no noise to select the first peak. In Fig. 7(e), some noise is added to test the algorithm. Comparison of several experimental results shows that the first peak of Fig. 7(d) is small, and it should be neglected because this type of peak can be caused by noise, and the method selects the third peak because the second (red) peak was already labeled in the previous calculation step.

Figure 8 shows results of EPI separation. The EPI and its successful separation into component images can be shown in the second and third columns. The second row shows that the proposed method can separate two layers even in the presence of specularly in both actual and virtual objects. However, in Fig. 8 (f), we can observe small artifacts on EPI strips that are

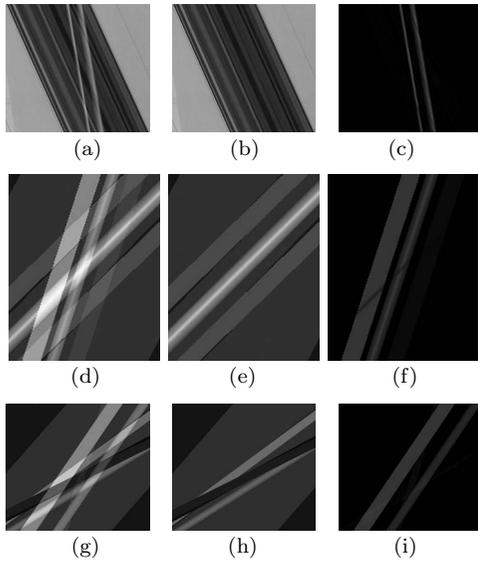


Fig. 8 Result of EPI analysis. (a) Input EPI of indoor image sequence. (d) Input EPI with view-dependent effect. (g) Input EPI with depth disparity. (b),(c),(e),(f),(h),(i) Separation results.

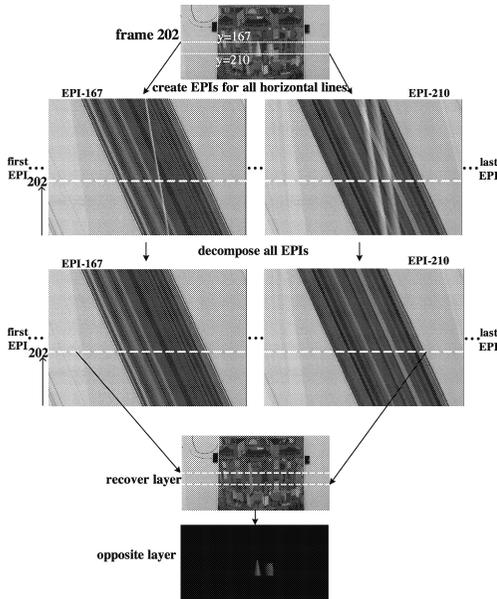


Fig. 9 Flow diagram of the iterative method for original image separation.

caused by color saturation on the synthesized image. To avoid such artifacts, using a high dynamic range image (HDRI) is a practical solution.

4.4 Separation of the Original Image

By using the result of the decomposition of EPI as described above, we can separate the original image into two component images by

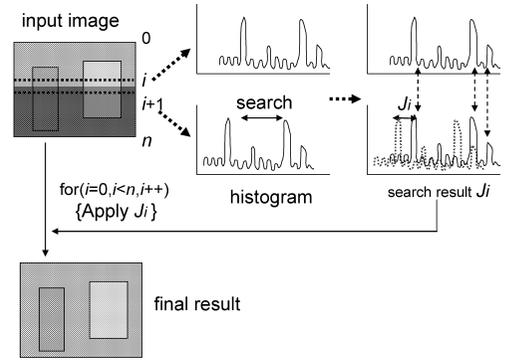


Fig. 10 The removal of color ambiguity.

two methods. The first, called the iterative method, is a straightforward method that creates EPIs for all horizontal lines and applies a separation algorithm to each EPI. The second method is based on color clustering.

Figure 9 is the diagrammatic explanation of the iterative method. This method produces artifacts between horizontal lines because there remains color ambiguity in each horizontal line of the recovered image. To remove this ambiguity, we apply the steps listed below. **Figure 10** is the diagrammatic explanation of this method.

- 1 Make histograms for all horizontal lines of a separated image.
- 2 for $i=0$ to height-1
 - a for $j=0$ to max color value
 - i. Compare top n peaks of histograms between horizontal line i and $i+1$.
 - ii. if $m(\leq n)$ or more number of peak values coincide

then exit loop and change flag to true.

else shift all peak values of line $i+1$ with j
 - b If flag is true, shift all colors on horizontal line $i+1$ with shift value j

In our experiment, we set $n = 10$ and $m = 7$. The detailed procedure of the second (color clustering) approach is as follows.

- 1 Select a horizontal line on the original image.
- 2 Create a sparse EPI from the captured image sequence that coincides with the selected horizontal line.

- 3 Decompose the EPI by the separation algorithm.
- 4 Perform region-growing method based on RGB color clustering starting from the selected horizontal line in the original image.
- 5 Retrieve the decomposed color for the clustered image from the decomposed sparse EPI.
- 6 Select the next horizontal line and iterate from 2 to 5 until all regions of the original image are decomposed.

Figure 11 is the diagrammatic explanation of the second approach.

Figure 12 shows the separation results of

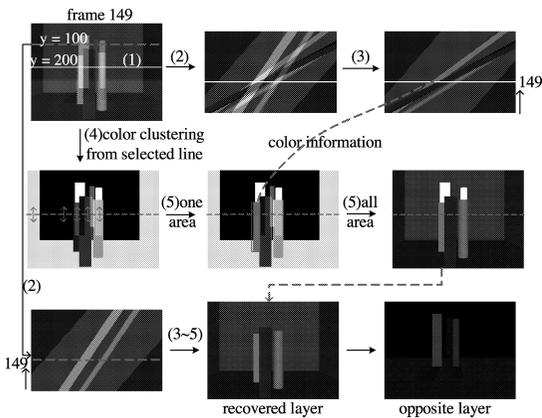


Fig. 11 Flow diagram of color clustering-based original image separation.

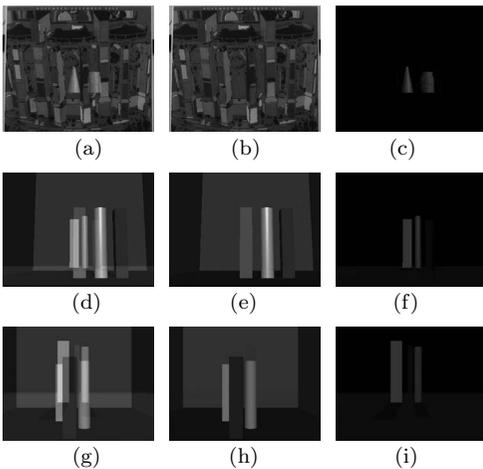


Fig. 12 The separation of original images. The first column shows the original input images. Separation results can be seen in the second and third columns.

original images. The left column of Fig.12 shows arbitrary frames of three input image sequences, and the recovered transparency and reflection component images are shown in the middle and right columns. For Fig.12 (a), the iterative method is used, because the image texture is complicated, and it is difficult to achieve an accurate color clustering result. The second (color clustering) method is applied for the separation of Fig. 12 (d) and (g), because of the simple texture and the fact that most EPIs have the same pattern and color information.

5. Experiments

We performed several experiments to test the effectiveness of our method. In the following two experiments, we used indoor image sequence captured in our laboratory and outdoor scenes.

5.1 Indoor Image Sequence

We have conducted several tests on real images captured using the Sony 3-CCD (640×480) digital camera. The motorized stage has been used to control the linear movement of the cam-



Fig. 13 The scene of the real image capturing process.

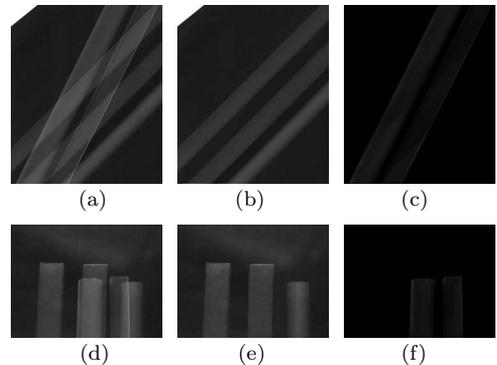


Fig. 14 The separation of indoor image sequence. The first row shows the input EPI and its separation results. The original image and separation results are shown in the second row.

era as shown in **Fig. 13**. **Figure 14** shows the experimental results using indoor image sequence with simple texture. The second (color clustering) separation method is used for the separation of original images of this image sequence. In Fig. 14, we can see that the input EPI is successfully separated into component images. However, for the separation of original images, we can observe some artifacts in

Fig. 14 (f). The main reason for this error can be considered as the non-linearity of the camera sensor. Therefore, using a truly calibrated camera is a desirable solution.

5.2 Outdoor Image Sequence

A car-mounted video camera has been used to

Table 1 Specification of outdoor experiments.

	Exp: 1	Exp: 2
Model No.	SONY VX-2000	SONY VX-2000
Shutter speed	1/180 sec	1/250 sec
F value	F8	F6.8
Frame rate	29.97 frame/sec	29.97 frame/sec
Image size	720 × 480 pixel	720 × 480 pixel
Location	University Campus	Expressway
Speed (approximate)	5km/h	60km/h

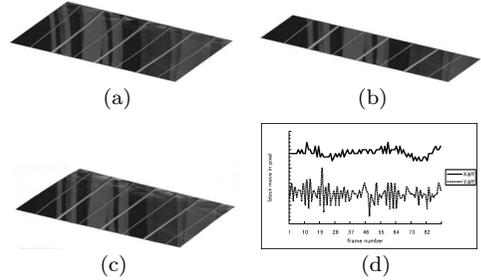


Fig. 15 Rectified EPIs: (a) using all EPI range (b) using reduced EPI range (c) fitted with block matching result (X_{diff}). (d) block matching result.

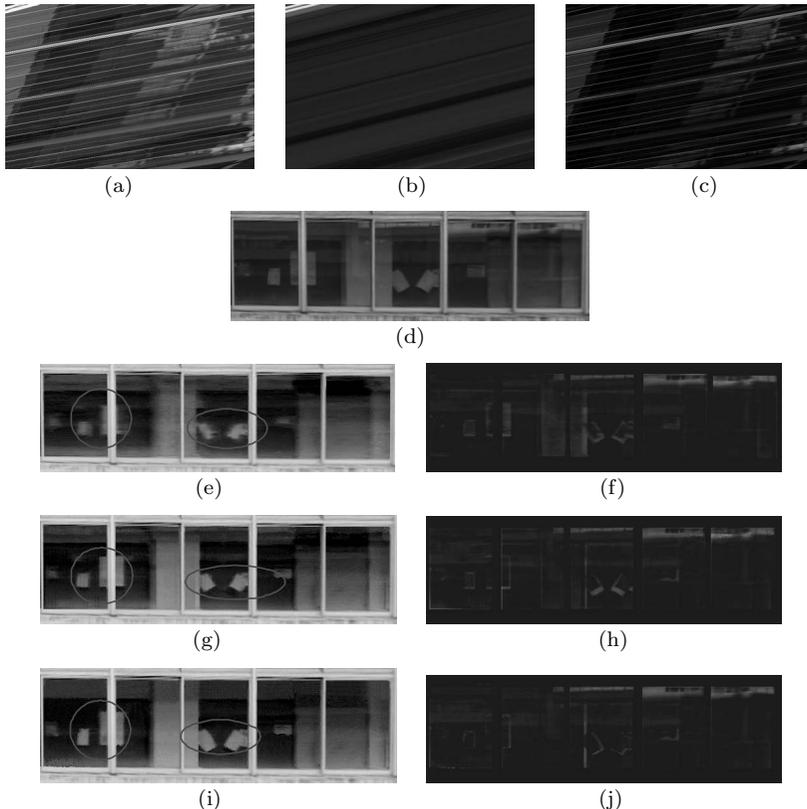


Fig. 16 Experimental results using outdoor image sequence 1: first row shows the input EPI and its separation results, second row shows the input outdoor image, third row shows the separation results using the entire EPI range. The results described in the fourth row are decomposed by using a reduced EPI range. The bottom row shows the results by using motion estimation data. The right column of the result images is enhanced to make them easier to see.

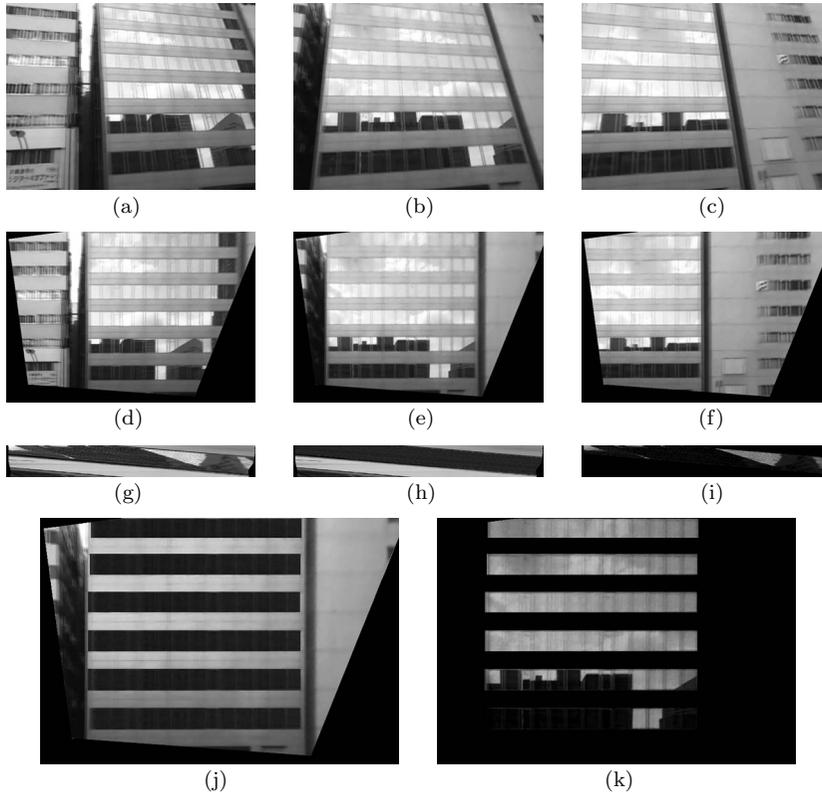


Fig. 17 Experimental results of outdoor image sequence 2: first row, captured image frames; second row, rectified image frames; third row, EPI and separation results; bottom row, separation result of (e).

capture outdoor image sequences by restricting the car to a constant speed and driving along a straight path. The detailed experimental conditions are described in **Table 1**. Although we tried to keep the car speed constant and driving along a straight way, a tiny lack of constancy still occurred between each successive image frame as describe in Section 4.1. Therefore, the trails within the EPI-strips are not strictly vertical as shown in **Fig. 15** (a). To solve this problem, we can simply apply the straightforward method, which is to reduce the number of image frames in EPI until the rectified strip appears as vertical as shown in Fig. 15 (b). Figure 15 (c) shows the rectified EPI, which is produced by adjusting the speed difference of the camera by the block matching algorithm as described in Section 4.1. In this paper, we provide the results using both methods and compare them. As a result, the proposed method can extract two component layers even when the image texture is complicated and even if we can observe specular effect on the reflected buildings.

Figure 16 shows the separation results of an image sequence. Figure 16 (a) is an example input EPI produced by accumulating line 26 of each image frame, and (b) and (c) are the EPI separation results. One image frame from the original input image sequence can be seen in the second row (d). The third row (e, f) shows the separation results using all EPI ranges, and the fourth row (g, h) shows results using short-term EPI. The bottom row (i, j) shows the results by using the block matching method. We can clearly see that the EPI is successfully decomposed in the first row (b, c), and consequently the original image is also decomposed effectively (e–j). In addition, both methods using short term EPI and the preprocessed method improved results drastically as shown in Fig. 16 (g–j). Red circles on Fig. 16 depict the object inside the glass; note that the half transparency is successfully separated with our method.

The experimental results of another outdoor image sequence can be seen in **Fig. 17**. In this experiment, we manually separate the tar-

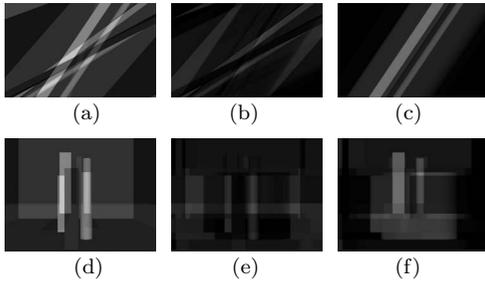


Fig. 18 The separation result of EPI and original image using Szeliski's method.

get building and the next building. However, to apply our technique to a large-scale scene, we should use an automatic method, e.g., Kawasaki and Ikeuchi's method⁶⁾. In Fig. 17, some arbitrary frames from an input image sequence are shown in first row. The second row shows the rectified images. Figure 17 (g) is an example input EPI produced by accumulating line 360 of each image frame, and (h) and (i) are the EPI separation results. The separation results of Fig. 17 (e) can be seen in the bottom row (j, k), and we can see that our algorithm has successfully extracted two layers from outdoor images.

5.3 Comparison to other method

To evaluate the applicability of our method, we compared our method to R. Szeliski's method¹⁴⁾, which is the only method assuming a moving camera. **Figure 18** shows the separation result of synthesized image sequence by their method. This is the same image sequence used in Figure 12, third row. Since their method assumes a single depth for each layer, we calculated average depth for both layers for the separation. The depth difference between objects was small in this experiment. However, their method could not recover the correct layer, and it produced totally different colors for all objects in the scene.

6. Discussion

In this paper, we proposed a method that can be applied not only for indoor but also for outdoor scenes. The main difference between an indoor and outdoor scene is whether or not we can control the environmental situations that contain variations in:

- (1) Material and 3D shape of the objects (e.g., glossy surface and intricate 3D shape)
- (2) Camera motion (e.g., speed, path)
- (3) Lighting conditions

- (4) Dynamic range (e.g., high)

In case of 1 to 3, our method worked successfully to separate reflectance and transparency, as shown in the experiment section. In terms of case 4, we have not tried to test a high dynamic range image (HDRI) in this paper. However, treating HDRI is currently common in Computer Vision (CV) and Computer Graphics (CG), and our method is also expected to work properly. We intend to test this in our future work.

7. Conclusion

In this paper, we propose an efficient method to separate reflection and transparency layers for outdoor image sequences. This is difficult to achieve because of large depth disparity in actual outdoor scenes, such as buildings in an urban location.

To solve this problem, we have analyzed the color and geometrical nature of the strips on the EPI and proposed a simple and robust method to separate the EPI into component layers. In addition, we also proposed a method to decompose original images into reflection and transparency images using EPI separation results.

To apply the technique to an outdoor environment, we have to compensate for irregular camera motion, and we provide two solutions: the short EPI method and the block matching method.

By using our color-based EPI analysis and camera motion compensation methods, a scene consisting of glass-like objects, which produce both a transparent and reflective effect, could be robustly separated into component images.

References

- 1) Bolles, R., Baker, H. and Marimont, D.: Epipolar plane image analysis: An approach to determining structure from motion, *International Journal of Computer Vision*, No.1, pp.7–55 (1987).
- 2) Bronstein, A.M., Bronstein, M.M., Zibulevsky, M. and Zeevi, Y.Y.: Blind separation of reflections using sparse ICA, *4th International Symposium on Independent Component Analysis and Blind Signal Separation*, pp.227–232 (Apr. 2003).
- 3) Farid, H. and Adelson, E.H.: Separating reflections from images using independent components analysis, *Journal of the Optical Society of America*, No.16, pp.2136–2145 (1999).
- 4) Hermanto, A., Barros, K., Yamamura, T. and Ohnishi, N.: Separating virtual and real objects

using independent component analysis, *IEICE TRANS*, E84-D (Sep. 2001).

- 5) Irani, M. and Peleg, S.: Motion analysis for image enhancement: resolution, occlusion, and transparency, *Journal of Visual Communication and Image Representation*, Vol.4, No.4, pp.324–335.
- 6) Kawasaki, H., Yatabe, T., Ikeuchi, K. and Sakauchi, M.: Automatic modeling of a 3d city map from real-world video, *In ACM Multimedia*, pp.11–18 (Oct. 1999).
- 7) Sarel, B. and Irani, M.: Separating transparent layers through layer information exchange, *ECCV*, Vol.4, pp.328–341 (May 2004).
- 8) Schechner, Y.Y., Kiryati, N. and Basri, R.: Separation of transparent layers using focus, *International Journal of Computer Vision*, No.39, pp.25–39 (2000).
- 9) Schechner, Y.Y., Kiryati, N. and Basri, R.: Separation of transparent layers using focus, *ICCV*, pp.1061–1066 (1998).
- 10) Schechner, Y.Y., Kiryati, N. and Shamir, J.: Separation of transparent layers by polarization analysis, *Scandinavian Conference on Image Analysis*, Vol.1, pp.235–242 (1999).
- 11) Schechner, Y.Y. and Shamir, J.: Vision through semireflecting media: polarization analysis, *Optics Letters*, No.24, pp.1088–1090 (1999).
- 12) Schechner, Y.Y., Shamir, J. and Kiryati, N.: Polarization-based decorrelation of transparent layers: The inclination angle of an invisible surface, *ICCV*, Vol.2, pp.814–819 (1999).
- 13) Swaminathan, R., Kang, S.B. and Szeliski, R.: On the motion and appearance of specularities in image sequences, *ECCV*, Vol.1, pp.508–523 (2002).
- 14) Szeliski, R., Avidan, S. and Anandan, P.: Layer extraction from multiple images containing reflections and transparency, *CVPR*, pp.1246–1253 (June 2000).
- 15) Yamamoto, M.: Determining 3-d structure of scene from image sequences obtained by horizontal and vertical moving camera, *Pattern Recognition*, pp.458–467 (1988).

(Received September 1, 2005)

(Accepted March 17, 2006)

(Editor in Charge: Yasuhiro Mukaigawa)



Thanda Oo received her B.Sc (Hons:) degree from Yangon University and M.I.Sc degree from Yangon University of Computer Studies, in 1994 and 1998, and Ph.D. degree from Saitama University in 2006 respectively. She has also worked at the Toshiba Information System Corporation.



Hiroshi Kawasaki received a B. Eng. degree in electrical and electronic engineering from Kyoto University, Kyoto, Japan, in 1994, and M.E. and Ph.D. degrees in Information and Communication Engineering from the University of Tokyo, Japan, in 2000 and 2003, respectively. He is an associate professor in the Department of Information and Computer Sciences faculty of engineering, Saitama University, Japan. His current research focus is on acquisition of dense 3D data and constructing photorealistic 3D scene models from multiple images and 3D data. He is a member of IEICE, IPSJ, and IEEE.



Yutaka Ohsawa received B.E. and M.E. degrees in Electronics from Shinsyu University in 1976 and 1978 respectively, and a Ph.D. degree in Computer Science from the University of Tokyo in 1985. In 1979, he joined the Institute of Industrial Science, the University of Tokyo, where he focused on image processing and image databases. In 1990, he moved to Saitama University where he is a professor in the Department of Information and Computer Sciences. His research interests are geographic information system and spatiotemporal information systems.



Katsushi Ikeuchi received a B. Eng. degree in mechanical engineering from Kyoto University, Kyoto, Japan, in 1973, and a Ph.D. degree in information engineering from the University of Tokyo, Tokyo, Japan, in 1978.

He is a professor at the Interfaculty Initiative in Information Studies, the University of Tokyo, Tokyo, Japan. After working at the AI Laboratory at the Massachusetts Institute of Technology for three years, the Electrotechnical Laboratory for five years, and the School of Computer Science at Carnegie Mellon University for 10 years, he joined the University of Tokyo in 1996. He was selected as a Distinguished Lecturer of the IEEE Signal Processing Society for the period of 2000–2001, and a Distinguished Lecturer of the IEEE Computer Society for the period of 2004–2006. He has received several awards, including the David Marr Prize in ICCV 1990, IEEE R&A K-S Fu Memorial Best Transaction Paper Award in 1998, and best paper awards in CVPR 1991, VSMM 2000, and VSMM 2004. In addition, in 1992, his paper, “Numerical Shape from Shading and Occluding Boundaries,” was selected as one of the most influential papers to have appeared in the Artificial Intelligence Journal within the past 10 years. He is a fellow of the IEEE.
