Outline of Three-Dimensional Image Processing

This paper assumes that three-dimensional image processing techniques are "techniques of processing images that have been taken by a number of cameras to create arbitrary-viewpoint images that can be viewed from various different directions". At NHK, we are working on developing tomorrow's highly realistic imaging techniques and innovative image production techniques, as techniques of processing three-dimensional image processing. Among these projects, we have developed various applications, such as those that combine techniques of measuring the configuration of a subject from images taken by a number of cameras, techniques that enable manipulation of a user's viewpoint based on that configuration, and augmented reality techniques.

1. Introduction

At NHK Science and Technology Research Laboratories (STRL), we are currently researching image processing techniques for generating arbitrary-viewpoint images that can be viewed from various different directions, using a number of cameras. We are mainly focusing on methods of generating dynamic three-dimensional models and the development of multi-viewpoint HDTV systems.

As part of the research into dynamic three-dimensional models, we are investigating methods of generating a three-dimensional model of a subject for each frame of footage, from multi-viewpoint footage that was taken by a number of cameras arranged around the subject. We can display arbitrary-viewpoint images of a moving subject by sequentially displaying three-dimensional models that have been reproduced in each frame of the footage, in a time series. This dynamic three-dimensional model enables a realistic reproduction of the detailed movements of human beings and even the movements of wrinkles in their clothing, and is expected to be usable as innovative image material in place of computer graphics (CG).

A multi-viewpoint HDTV system implements an image representation method called "bullet time" in broadcasts such as sports relays. This system has a number of cameras arranged in a line or in a semi-circular arc with respect to the subject, and the images from the cameras are switched in sequence, to enable the creation of images as if a single camera is moving. If the switching of the images is not smooth, the result would be ugly footage, so this system performs image processing at high speed in order to switch the images smoothly. Since this makes it possible to comment on player movements from a variety of viewpoints in a broadcast program such as a sports program, it enables rendering that is readily comprehensible and effective. This system is configured of a PC for recording from and controlling 12 HDTV cameras, to create a system that emphasizes operability and immediacy. This has already been used in live broadcasts such as gymnastics shows.

In this paper, the basis of three-dimensional image processing techniques will be described first. Also, the reproduction of surface features of Mars and a system that uses augmented reality will be introduced, as examples of related research topics.

2. Fundamentals of three-dimensional image processing techniques

2.1 Camera modeling

To enable three-dimensional image processing, it is first necessary to represent each camera as a mathematical expression. In general, a camera consists of a lens and an imaging plate, but in this case we use a pinhole camera model which is the simplest type of camera. The principle of the pinhole camera is shown in Figure 1(a). Light from the subject passes through the pinhole and an inverted image is projected onto an image plane (ground glass) at the rear. With a pinhole camera, there is no concept of focal length, but for descriptive purposes, we call the distance between the pinhole and the image plane the focal length. To further simplify the configuration, the image plane is moved to in front of the pinhole, as shown in Figure 1(b). If an erect image is projected onto the image plane at the rear, a pinhole camera can be represented consistently. Note that the pinhole is equivalent to the central position of a lens, and this position is called the optical center.
To represent the camera model of Figure 1(b) as a mathematical expression, we define a coordinate system such as that shown in Figure 2. Figure 2(a) shows the state in which a point $P$ in space is projected by the camera model onto a point $p$ in the image plane. The camera coordinates are three-dimensional coordinates based on an optical center $O_c$. The image plane is positioned at just the focal length $f$ in the $z$-axis direction of the camera coordinates, and the $u$-$v$ plane of the image coordinate system is parallel to the $u$-$v$ plane of the camera coordinate system. The $z$-axis of the camera coordinate system intersects the image plane at $s (C_u, C_v)$ in the image coordinates. The $z$-axis in the camera coordinates is the direction in which the camera captures images, and is also called the optical axis.

Figure 2(b) is the view from directly above Figure 2(a). The point $P(x, y, z)$ is projected onto the point $p(u, v)$ of the image plane. Since triangle $P-S-O_c$ is similar to triangle $p-s-O_c$, Equation (1) holds:

$$\begin{pmatrix} (u - C_u) \Delta_u \\ (v - C_v) \Delta_v \end{pmatrix} = \frac{f}{z} \begin{pmatrix} x \\ y \end{pmatrix}$$

(1)
where $\Delta_x$ and $\Delta_y$ are pixel dimensions. Transforming Equation (1) gives Equation (2):

$$\begin{bmatrix} x - C_x \\ y - C_y \end{bmatrix} = \begin{bmatrix} 1/\Delta_y & 0 & 0 \\ 0 & 1/\Delta_x & 0 \end{bmatrix} \begin{bmatrix} a \cdot f \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

where $a = \Delta_y/\Delta_x$ is the pixel aspect ratio. In addition, $F$ is used as the focal length $f$ normalized by $\Delta_x$, to transform Equation (2) and obtain Equation (3):

$$\begin{bmatrix} u \\ v \\ z \end{bmatrix} = \begin{bmatrix} a \cdot F & 0 & C_x \\ 0 & F & C_y \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Equation (3) describes the projection of a point within the space defined by the camera coordinates onto the image plane, where the 3x3 matrix on the right side is a perspective projection matrix. The matrix of Equation (3) is that of an ideal camera, and components such as those for lens distortion can also be added. Parameters such as the focal length comprised within the perspective projection matrix are camera-specific, and so are called internal parameters.

2.2 Global coordinates

Global coordinates are a coordinate system that represents the entire space, which is used for defining the position and orientation of the cameras. As shown in Figure 3, the conversion from global coordinates to camera coordinates can be done by rotation and parallel transition. Coordinate conversion makes it possible to project a point that has been defined in global coordinates onto the image plane of a camera.

The equation for converting global coordinates into camera coordinates is:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} R & T \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

where $R$ is a rotation from global coordinates to camera coordinates and $T$ is a parallel transition from the origin of the camera coordinates to the origin of the global coordinates. Since $R$ and $T$ are parameters that do not depend on the camera’s specifications, they are called external parameters. If the perspective projection matrix is represented by $A$ and Equations (3) and (4) are combined, we get Equation (5) that projects a point in space that has been defined in global coordinates onto the image plane.

$$\begin{bmatrix} u \\ v \\ z \end{bmatrix} = A \begin{bmatrix} R & T \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

2.3 Camera calibration

The parameters $A$, $R$, and $T$ of Equation (5) are obtained by camera calibration. Using matrix components to substitute into Equation (5) gives Equation (6):

$$\begin{bmatrix} u \\ v \\ z \end{bmatrix} = \begin{bmatrix} f_{11} & f_{12} & f_{13} & f_{14} \\ f_{21} & f_{22} & f_{23} & f_{24} \\ f_{31} & f_{32} & f_{33} & f_{34} = 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

If we substitute the image coordinates and global coordinates of one characteristic point into Equation (6) to eliminate $z$, two equations that correlate the image coordinates $u$ and $v$ are obtained. The right side of this equation is a fractional expression in which $z$ is
denominator, and universality is not lost even if one matrix component $s_j$ is set to 1. With Equation (6), setting $s_{34}$ to 1 results in 11 unknowns $s_{11}$ to $s_{33}$. To derive the 11 unknowns, we create equations for at least six characteristic points and solve them as simultaneous equations. The terms $s_{11}$ to $s_{34}$ can be solved in $A$, $R$, and $T$ by using the property that the three column vectors of the rotational matrix $R$ intersect each other.

To perform this kind of camera calibration, it is enough to know the correspondence between the global coordinates and image coordinates of the characteristic point. For that purpose, a pattern such as that shown in Figure 4 could be used.

The example of Figure 4 is configured of two plane patterns that correspond to the $X$-$Z$ plane and the $Y$-$Z$ plane in the global coordinate system. In each of the two planes, a rectangular grid pattern is drawn uniformly, and the vertices of the grids are used as characteristic points. The global coordinates of each vertex are known, so the image coordinates are obtained by extracting the vertices of the grid from the image of the pattern.

2.4 Distance measurement by stereo camera

In this chapter, the measurement of distances by a stereo camera is discussed, as a typical example of a three-dimensional image processing technique.
representation of a stereo camera by the pinhole camera model is shown in Figure 5. The focal length of the camera is $f$, the width of the image plane is $W$, the pixel spacing is $\Delta_x$, the camera spacing is $B$, and the distance from the camera to the subject is $z$. The tree that is the subject is projected onto the positions of the $u_L$-th pixel of the left camera and the $u_R$-th pixel of the right camera, and the spacing between the two projection points is $x$.

When the position of the tree is assumed to be a common vertex, the triangle having a bottom edge of $x$ and the triangle having a bottom edge of $B$ are similar, so that Equation (7) holds:

$$\frac{z}{B} = \frac{z - f}{x}$$

Thus, the distance $z$ from the camera to the tree is given by Equation (8):

$$z = \frac{B}{B - x} f = \frac{B}{d - \Delta_x} f$$

It should be noted that $d = u_L - u_R$. $\Delta_x$ is obtained from the camera specifications, and $B$ and $f$ are obtained by camera calibration. Thus, $z$ can be calculated if the amount $d$ that is called disparity is obtained. To obtain $d$, it is necessary to obtain the correspondence between the images of the left and right cameras. In the example shown in Figure 5, if there is a pixel that appears in the portion of the tree captured by the left camera, a pixel for the same part of the tree can be seen from the right-camera’s image. Such processing is called a congruent points search. Performing a congruent points search for all pixels of the stereo images provides the disparity for each pixel. As will be described later with reference to Figure 7, images in which the amount of disparity is represented by brightnesses are called disparity images.

2.5 Visual volume intersection

The visual volume intersection method is one method of generating a three-dimensional model of a subject from images taken by a number of cameras surrounding the subject. A directly overhead view of three cameras and the subject is shown in Figure 6. The central grid is a vertex space that is defined in global coordinates, such as a three-dimensional grid with vertices spaced 1-cm apart. The yellow region within the vertex space is the actual shape of the subject and the white portion within the image plane of each camera is the silhouette of the subject.

As can be seen from Figure 6, each camera creates a shape like a triangle. Spatially, it forms a cone in space with the vertex at the optical center of the camera and a cross-section that is the silhouette shape. Such a cone is called a visual volume. The actual form of the subject is somewhere within these cones. If a number of cameras are used, a number of visual volumes are formed. The region where these visual volumes intersect is the visual volume intersection region (Visual Hull), and this region denotes the approximate shape of the subject.

To obtain the visual hull, we project the vertices of the vertex space onto silhouette images of a number of cameras, and determine whether or not each projected
point is within the corresponding silhouette region. First of all, the vertex coordinates are substituted into the \((X, Y, Z)\) terms on the right side of Equation (5), to obtain the image coordinates \((u, v)\) of the projection point. The silhouette is determined in the projection points. If a projection point is included in the silhouette regions of all the cameras, that vertex is determined to be within the subject. This determination is done for all the vertices of the vertex space, to obtain the visual hull.

The visual hull is a shape that circumscribes the actual shape. Increasing the number of cameras improves the approximation accuracy, but concave parts of the surface of the subject are not shown in the silhouettes so cannot be reproduced. The visual volume intersection method does have this fundamental defect, but it is a stable method that does not generate major errors. To remedy that defect, a method such as that using a stereo camera that was described in Section 2.4 (stereo matching method) could be used in combination to improve the accuracy of shape reproduction.

### 3. Reproducing the features of the Martian surface

We have developed a method of generating surface features from stereo images taken of the surface of Mars, with the objective of using them in NHK’s HDTV special program called: ”Space Romance: The Mysteries of the Solar System” (broadcast April 2006). Once the surface features have been obtained, it becomes possible to create images in which the angle can be freely changed, and also achieve a variety of rendering effects.

#### 3.1 Stereo images of Mars

The Mars images that we used were panoramic stereo images taken by NASA’s Mars Exploration Rover, Spirit, and satellite images taken by the European Space Agency’s Mars exploration mission, Mars Express. The panoramic stereo images taken by Spirit are stereo images taken while the panoramic camera was panned, and the images taken from the satellite are pairs of images taken from different positions on the same orbit.

#### 3.2 Congruent points search

As shown in Figure 7(a), the stereo images of the Martian surface have many areas where there are few changes in brightness, making it difficult to obtain an accurate equivalence by the congruent points search which relies on changes in brightness as clues. In such a case, we perform census conversion as described below on the left and right stereo images, and use a method that performs a congruent points search between the census images.

A census image \(C_{(x,y)(i,j)}\) is an image in which only values for surrounding pixels are stored as shown in Equation (9), where the brightness value of image coordinates \((x, y)\) is set to \(I_{(x,y)}\), the magnitude of that value is compared with the brightnesses of \(N\) peripheral pixels \(I_{(x+i, y+j)}\). This is set to 1 if the brightness value of a surrounding pixel is greater than or equal to that of the image coordinates, or it is set to 0 if it is less than that.

\[
C_{(x,y)(i,j)} = \begin{cases} 
1 & \text{if } I_{(x,y)} \leq I_{(x+i, y+j)} \\
0 & \text{else} 
\end{cases} 
\]  

(9)

It should be noted that \(C_{(x,y)(0,0)}\) is normally 1 and holds no information. For instance, if the peripheral region \(N\) is assumed to be 3x3 pixels, \(C_{(x,y)}\) holds information about an 8-bit comparison result. We subject the left and right stereo images to census conversion to obtain two census images \(C^L\) and \(C^R\). We perform a congruent points search based on the Hamming distance between those census images, and obtain a disparity \(d\) such that the evaluation function of Equation (10) is at a minimum.

\[
H(d) = \sum_{i,j \in N} C^L_{(x,y)(i,j)} \otimes C^R_{(x+i, y+j)} 
\]  

(10)

where \(\otimes\) is an exclusive OR. Even when using census images, regions that have few changes in brightness could result in the creation of erroneous correspondences in image regions over a wide range. If an erroneous correspondence occurs, noise-like irregularities will be created when the surface features are generated. In such a case, we assume that ‘the disparity of adjacent pixels will change smoothly’ to stabilize the congruent points search. In other words, we use an evaluation function to which a term is added to \(H(d)\) for stabilization. The minimum value of such an evaluation function can be obtained by iteration calculations, and the evaluation function for the \(t\)-th iteration is given by Equation (11):

\[
E(d_t) = H(d_t) + \lambda \cdot R(d_t) 
\]  

(11)

where \(R(d_t)\) denotes the stabilization term, \(d_t\) denotes the \(t\)-th amount of disparity, and \(\lambda\) is a coefficient for adjusting the degree of stabilization. \(R(d_t)\) is a function that expresses the difference between the amounts of disparity of the subject pixel \((x, y)\) and peripheral pixels as shown in Equation (12).

\[
R(d_t) = \sum_{(x,y) \in N_R} |d_{(x,y)} - d_{(x+i, y+j)}| 
\]  

(12)

where \(N_R\) denotes a pixel in the peripheral region and
Figure 7: Disparity images

(a) Captured image
(b) disparity image (zeroth iteration)
(c) Disparity image (first iteration)
(d) Fourth iteration plus low-pass filter processing

* The dark portions in the disparity images of (b) to (d) show disparity minima and the light portions show disparity maxima.

Figure 8: Images generated from surface feature data
the amount of disparity \(d_t\) that was obtained by the \((t-1)\)-th iteration is used in the disparity of the peripheral regions. We obtain the disparity that minimizes the evaluation function of Equation (12), then repeat the iteration processing while updating with the disparity \(d_t\) of the subject pixel. Finally, a low-pass filter is used to perform smoothing processing on the disparity image configured of disparity values obtained by this method.

### 3.3 Generation of Martian surface features

An example of the application of this method to stereo images that were taken by the Mars exploration mission Mars Express is shown in Figure 7. In Figure 7(b), the number of iterations is zero and the disparity is obtained with the second item on the right side of Equation (11) being zero. For that reason, noise-like disparity estimation errors can be seen. With the first iteration shown in Figure 7(c), the noise-like disparity estimation errors are reduced.

If triangular patches are applied to the disparity image of Figure 7(d) and the result is converted into three-dimensional data, images with changing angles can be generated. That part is shown in Figure 8. This formation is called the "Hourglass", which is said to have been formed when ice on the mountain tops melted and the accumulated water flowed out all at once. Note that the disparity image does not represent shapes, but we assume that the satellite was at an altitude that was sufficiently higher than any variation in height of the land surface, so the disparity image can be used to approximate the surface features.

An example of the application of this method to the panoramic stereo images taken by the Mars exploration rover is shown in Figure 9. Figures 9(a) and 9(b) are panoramic stereo images. It is clear that these images are a montage of images taken while the cameras were panning. The two cameras of the rover's stereo camera setup are placed 30 cm apart. Panoramic stereo images are generated by linking together images that were taken while the cameras where panning horizontally. The generated disparity image is shown in Figure 9(c).

In the disparity image of Figure 9(c), disparity values are represented by brightnesses, where disparity is small in dark portions but large in bright portions. Using Equation (8), the disparity can be converted into the distance from the camera to the subject. If this calculation is done for each pixel, the form of the subject can be expressed by image coordinates \((u, v)\) and distance \(z\). However, since the original images are panoramic
images taken around the full 360°, the shapes of the subjects could not be represented correctly in the past. In this case, we use Equation (13) to convert the images into cylindrical shape.

\[
P_z = z \cdot \cos \left( \frac{\mu}{U} \cdot 360 \right) \\
P_r = z \cdot \frac{\Delta_x \cdot \nu}{f} \\
P_t = z \cdot \sin \left( \frac{\mu}{U} \cdot 360 \right)
\]

where \( U \) is the number of pixels in the lateral direction of the image. An example of the conversion into a cylindrical shape model is shown in Figure 10. On the left side of the figure shows the entire shape model, with the stereo camera positioned in the center. The right side of Figure 10 is an expanded part of the model, together with a wire-frame model of that part. The outer periphery of the shape model is shaped like a wall to ensure that the amount of data of the shape model is reduced and thus an upper limit is set on the distance values.

Figure 10 shows a region which was sea in the past. In this program, we reproduced the Martian sea by combining a transparent blue overlay with this shape model. Part of the reproduced video of images taken while the camera panned over this Martian sea is shown in Figure 11.
4. Virtual puppet system

The virtual puppet system integrates images of a three-dimensional model that are linked to life-action images. A card on which a special pattern is drawn is filmed by video camera; type, position, and orientation information on the card is detected from the filmed images, and a three-dimensional model is integrated onto the card and displayed. A tool for building augmented reality systems, called the Augmented Reality Tool Kit (ARToolKit), was used for card recognition. As shown in Figure 12, a black-and-white rectangular marker is drawn on the card, and the orientation and position of the card is deduced as described below.

We define the orientation of the card by three-dimensional coordinates created from two vectors that intersect within the surface of the card and a vector that is normal to the card. The sequence of these calculations is described below. An image that was taken of the card placed on white paper is shown in Figure 12(b). The outline of the rectangular marker in the filmed image is followed and the image coordinates of the four vertices A to D are obtained by detecting bends in the lines. From the image coordinates of the four vertices, we can express each edge as a linear equation. In this case, the linear equations for two facing edges are given by:

\[
\begin{align*}
\alpha_{\text{AB}} \cdot u + \beta_{\text{AB}} \cdot v + 1 &= 0 \\
\alpha_{\text{CD}} \cdot u + \beta_{\text{CD}} \cdot v + 1 &= 0
\end{align*}
\]  

(14)

where \((u, v)\) are the image coordinates, and \(\alpha\) and \(\beta\) are linear parameters. Substituting Equation (14) into Equation (3) gives Equation (15):

\[
\begin{align*}
\alpha_{\text{AB}} \cdot \alpha_{\text{F}} \cdot x + \beta_{\text{AB}} \cdot \beta_{\text{F}} \cdot y + (\alpha_{\text{AB}} \cdot C_y + \beta_{\text{AB}} \cdot C_x + 1)z &= 0 \\
\alpha_{\text{CD}} \cdot \alpha_{\text{F}} \cdot x + \beta_{\text{CD}} \cdot \beta_{\text{F}} \cdot y + (\alpha_{\text{CD}} \cdot C_y + \beta_{\text{CD}} \cdot C_x + 1)z &= 0
\end{align*}
\]  

(15)

Equation (15) gives the equations of the plane defined in camera coordinates \((x, y, z)\). The upper equation is the plane determined by the edge \(\text{AD}\) and the origin of the camera coordinates and the lower equation is the plane determined by the edge \(\text{BC}\) and the origin of the camera coordinates. The normal vector of each plane is \((\alpha_{\text{AB}} \cdot \alpha_{\text{F}} \cdot \beta_{\text{AB}} \cdot \beta_{\text{F}}, \alpha_{\text{AB}} \cdot C_y + \beta_{\text{AB}} \cdot C_x + 1)\) and \((\alpha_{\text{CD}} \cdot \alpha_{\text{F}} \cdot \beta_{\text{CD}} \cdot \beta_{\text{F}}, \alpha_{\text{CD}} \cdot C_y + \beta_{\text{CD}} \cdot C_x + 1)\). Since the two edges \(\text{AD}\) and \(\text{BC}\) are parallel, they have a common direction vector. This direction vector is calculated as the cross product of the normal vectors of the two planes of Equation (15). Two mutually intersecting vectors parallel to the card are obtained by doing these calculations for edges \(\text{AB}\) and \(\text{BC}\) as well. A vector perpendicular to the surface is obtained from the cross product of these two vectors. These three vectors represent the orientation of the marker with respect to the camera coordinate system. In addition, a 3x3 matrix in which these three vectors are column vectors represents the rotational matrix \(R\) from the global coordinates to the camera coordinates.

The position of the card is the parallel transition component \(T\) of Equation (5). The internal parameter \(A\) is measured beforehand and the rotational matrix \(R\) is known. Eight equations are obtained by substituting the marker coordinate system \((X, Y, Z)\) and the image coordinate system \((u, v)\) for the four vertices A to D into Equation (5). We obtain a parallel transition component by solving these equations by the least-squares method.

We can draw a three-dimensional model based on the information on the orientation and position of the card obtained in this way, then incorporate a three-dimensional model as if it is standing up on that card by integrating it into the camera images. Note that the pattern drawn on the central part of the card identifies the type of card.
4.1 Basic system for virtual puppet

The basic system configuration for virtual puppets is shown in Figure 13, together with a view of the integrated display. The system is configured of a single PC that performs processing such as importing camera image, recognizing the card, drawing the three-dimensional model, and integrating it into the camera image. Three-dimensional models of a number of characters are stored in the PC, with each three-dimensional model corresponding to a different type of card. Since the three-dimensional model is drawn with reference to the orientation information of the card, the three-dimensional model is always drawn as if it is standing on the card. This card recognition and integration is done in real time, so that the three-dimensional model rotates and moves as the user rotates and moves the card. This makes it possible to display a three-dimensional model from a number of viewpoints interactively.

To produce a high-quality image of the virtual puppet, the system has been converted to HDTV. To enable processing in real time, the configuration has been split between two PCs that do card detection processing and three-dimensional model drawing and integration processing. The configuration of this HDTV virtual puppet system is shown in Figure 14. Card detection is done by PC1. The type, position, and orientation of the detected card are passed through the network to PC2, where drawing and integration of the three-dimensional model is done at HDTV resolution.

![Figure 13: Basic system configuration for virtual puppet and view of integrated display](image1)

![Figure 14: HDTV virtual puppet system](image2)
4.2 Deduction of position and orientation of palm of hand

With ARToolKit, a clever trick is used to enable simple identification of the characteristic points of the marker, by using a black-and-white rectangle. If a rigid material is used for the card, there is little deformation and it is possible to deduce the orientation of the card stably. However, it can be assumed that the card could show within the integrated image, to produce an unwelcome effect. That is why we have investigated a method of using the palm of the hand instead of a card.

A template image that acts as reference for orientation is shown in Figure 15(a), showing the image of the palm of a hand with the fingers spread wide, as seen from the
We deduce the orientation and position of the palm of the hand in Figure 15(b) by obtaining a Homography matrix from Figure 15(a) to Figure 15(b), using characteristic points at the fingertips and wrist (circles) as clues.

The flow of processing for detecting the fingertips is shown in Figure 16. First of all, the system identifies the skin color, and obtains a silhouette of the palm of the hand, as shown in Figure 16(a). Next, the silhouette image is approximated by being bounded by a polygon, as shown in Figure 16(b), and the group of vertices of the polygon become candidates for fingertips. These candidate points include parts of the sleeve that the person is wearing. Since we want to detect just the fingertips, the system checks the silhouette shape in the vicinity of candidate points and assumes that candidate points in regions that are well-rounded are fingertips. As a result, the fingertip characteristic points can be extracted, as shown in Figure 16(c). We also concentrate on the distances between the five characteristic points and create equivalences between the characteristic points and the different fingers.

As will be described later, if at least four characteristic points can be obtained from filmed footage, the Homography matrix for the template image can be calculated. However, since it is also possible that the hand would not be seen as a plane if only the fingertip characteristic points were used, we also included characteristic points at the wrist. In Figure 16(b), characteristic points have been obtained for the sleeve parts, but they are not used as characteristic points because the position of the sleeve is variable. In that case, the wrist position is detected again from the features of the silhouette outline of the palm of the hand. The blue circles in Figure 15(b) show the thus-obtained seven characteristic points.

Next, Homography matrix is obtained from those characteristic points. The template image of Figure 15(a) is assumed to be the X-Y plane in the global coordinate system. The equation for projecting from the X-Y plane into the camera image plane is obtained by substituting $Z = 0$ into Equation (16).

$$
\begin{bmatrix}
u \\
v \\
1
\end{bmatrix} =
\begin{bmatrix}
s_{11} & s_{12} & s_{14} \\
s_{21} & s_{22} & s_{24} \\
s_{31} & s_{32} & s_{34} = 1
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
1
\end{bmatrix}
$$

(16)
where matrix components are replaced as shown by Equation (17):

\[
\begin{bmatrix}
  u \\
  v \\
  1
\end{bmatrix} =
\begin{bmatrix}
  h_{11} & h_{12} & h_{13} \\
  h_{21} & h_{22} & h_{23} \\
  h_{31} & h_{32} & 1
\end{bmatrix}
\begin{bmatrix}
  X \\
  Y \\
  1
\end{bmatrix}
\]

Equation (17) gives the Homography from the template image defined by global coordinates to the camera image. In this equation, there are eight unknowns \( h_{11} \) to \( h_{32} \). If correspondences can be found for at least four characteristic points between the template image and the camera image, those unknowns can be obtained. A matrix that is configured with the term \( h \) is called a Homography matrix. To obtain the orientation and positional information of the palm of the hand, we utilized the ARToolKit functions. We set a rectangular marker in a virtual manner on the template image instead of on card, as shown in Figure 17, and convert the four corners of the marker by Equation (17). We obtain orientation and positional information by adapting the image coordinates of the converted four vertices into Equations (14) and (15).

An integrated image with a CG drawing is shown in Figure 18. The determination of skin color and the extraction of characteristic points might be unstable, but we found it possible to integrate a CG element that follows the movements of the palm of the hand.

5. Conclusions

In this paper, I mainly introduced examples of the usage of three-dimensional image processing techniques in the production of images, but we can anticipate that the techniques that have been developed here will go on to promote the development of highly realistic images in the future. In particular, the use of techniques for measuring the shape of subjects in 3D TV will give rise to the advantage that it will be possible to generate a variety of forms of 3D images, such as integral 3D TVs or lenticular type stereoscopic images based on that shape information. We would like to look forward to progress in the innovative development of research by the fusion of three-dimensional image processing techniques and 3D TV technology.

(Yuichi Iwadate)