

## Painting Robot with Multi-Fingered Hands and Stereo Vision

Shunsuke Kudoh  
The University of Tokyo  
Tokyo, Japan  
kudoh@cvl.iis.u-toyo.ac.jp

Koichi Ogawara  
The University of Tokyo  
Tokyo, Japan  
ogawara@cvl.iis.u-toyo.ac.jp

Miti Ruchanurucks  
The University of Tokyo  
Tokyo, Japan  
miti@cvl.iis.u-toyo.ac.jp

Katsushi Ikeuchi  
The University of Tokyo  
Tokyo, Japan  
ki@cvl.iis.u-toyo.ac.jp

**Abstract**—In this paper, we describe a painting robot with multi-fingered hands and stereo vision. The goal of this study is for the robot to reproduce the whole procedure involved in human painting. A painting action is divided into three phases: obtaining a 3D model, composing a picture model, and painting by a robot. In this system, various feedback techniques including computer vision and force sensors are used. As experiments, an apple and a human silhouette are painted on a canvas using this system.

### I. INTRODUCTION

Research into humanoid robots generates great interest in both the scientific community and popular culture. Ideally, a humanoid robot could perform a cooperative task with a human [1] or work in a dangerous area such as a construction site instead of a human [2], [3]. However, most of studies about humanoid robots today focus only on simple whole-body motion such as walking [4], [5], [6]. In order to enable a humanoid robot to perform useful work for a human, it is necessary that it can manipulate equipment as well as walk around freely. Therefore, manipulation by hands is an important topic in research related to humanoid robots.

In this paper, we describe a painting robot with multi-fingered hands and stereo vision that uses a paintbrush to create a picture. Because a brush is deformable, it is a challenging task to control it in an appropriate way to paint a picture. It is necessary to develop several manipulation methods with feedback control using computer vision and force sensors. In this study, the goal is to reproduce the entire procedure involved in human painting by a robot with a vision system and multi-fingered hands. We divide a painting action into three phases as shown in Figure 1: obtaining a 3D model, composing a picture model, and painting by a robot.

In the first phase, the 3D shape is reconstructed from multi-view range images or multi-view camera images. In this study, we do not aim to produce photorealistic painting in which two-dimensional appearance is copied as is, but aim for a system in which the object is analyzed from various viewpoints and its features are represented on a canvas. For this purpose, it is necessary to recognize the object as a 3D model.

In the second phase, an interpretation of the 3D model is performed in order to decide how to represent features of the object. A picture model is composed for describing the interpretation. In this paper, two methods are implemented to

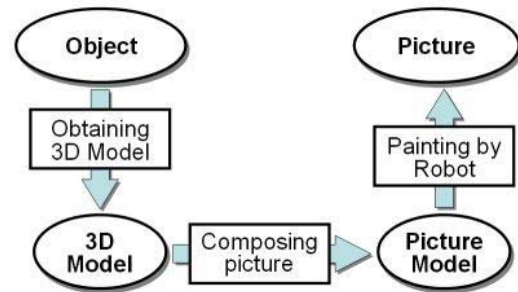


Fig. 1. Overview of the system

abstract features of the object: a method for representing an object by feature curves such as a silhouette that is extracted on the basis of geometrical edges, and a method for filling regions in a picture considering gradient in a texture of the object.

In the last phase, painting by a robot with multi-fingered hands is realized. One of the important challenges here is manipulation of a paintbrush by a multi-fingered hand. It is definitely different from drawing by an XY-plotter in which a paintbrush is fixed on the arm by bolts. In grasping by a multi-fingered hand, we should overcome problems such as the fact that the state of the grasp changes with motion and the grasp becomes unstable against a particular direction of force. The reason why we use a multi-fingered hand in spite of the difficulty is that we believe fingers are necessary to reproduce precise human techniques for manipulating a paintbrush. In this phase, visual feedback using stereo cameras is also performed. Because a brush deforms in painting, it is necessary to modify the picture by comparing the picture and the picture model. Through this operation, the robot gradually completes a picture by modifying it from observation as humans do.

In most of studies about manipulating objects by a robot, the objects are fixed on the robot arm and they study only the relationship among the objects. Some researchers have tried manipulation by a multi-fingered hand [7], [8], [9]. They classified grasp and developed an algorithm to manipulate objects based on the classification. However, they expected that their method would be used for a manufacturing application, and therefore few studies have been executed about manipulating daily objects, which are often deformable like

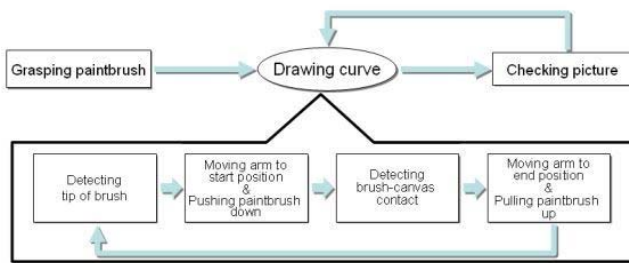


Fig. 2. Flowchart of painting by a robot

a brush in this study.

With regard to painting by a robot, Yokoyama et al. developed a painting robot for reproducing experienced painter's arts [10]. However, that study focused on sensing technology to obtain an expert's technique precisely by amgnetic sensors with a distortion correction algorithm, and painting performed by feed-forward control using a paintbrush fixed on a robot arm. We focus on painting by feedback control using a multi-fingered hand with force sensors and a stereo vision system.

In the remainder part of this paper, first, our robot hardware is introduced and the algorithm for painting a picture by the robot is described. Next, how to compose a picture model for the robot is described. Because of lack of space, reconstruction of a 3D shape is not described here, but its detail is described in [11]. We can suppose that 3D models of objects are correctly obtained in some appropriate way. Finally, experimental results of painting by the robot are shown.

## II. PAINTING BY ROBOT

The flowchart of painting by a robot is shown as Figure 2. First, the robot finds a paintbrush set up on the table and grasps it. Next, it enters the loop of drawing. In each loop, it recognizes the tip of the brush, moves the starting position of a curve, pushes the brush down to a canvas, and draws a curve. Finally, it compares the painting result with the picture model, and modifies the picture if necessary.

### A. Hardware

In this system, a robot shown in the left picture of Figure 3 is used. This robot has two robot arms with seven degrees-of-freedom (DOFs) each, and multi-fingered hands with four fingers. The first finger has four DOFs and the other fingers have three DOFs each. We use a multi-fingered hand in order to realize delicate technique of brushing like humans.

The robot also has a stereo vision system with nine cameras. It is used for obtaining the object's position in 3D space. The camera head equips a pan-tilt structure and can swing its head vertically and horizontally.

### B. Grasping Paintbrush

In this system, an ordinary commercial paintbrush is used. However, because the fingers of the robot are much thicker than the fingers of humans, a thick handle is attached to

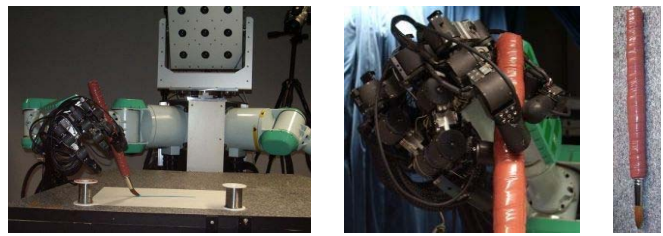


Fig. 3. Painting robot (left), grasping a handle (middle), and paintbrush for the robot (right)



Fig. 4. Process of grasping: the position of a paintbrush is detected by the vision system (left), the hand approaches the paintbrush, and the power of the fingers is gradually added (right).

the brush (Figure 3 right). In grasping, the paintbrush is supported by four points, which are three fingers (thumb, forefinger, and second finger) and the root of the thumb (Figure 3 middle). This is almost in imitation of human grasping of a paintbrush. This grasping produces stable support of a paintbrush against forces from all direction except one direction. The one direction occurs when the paintbrush moves from the far side to the near side. In order to avoid the instability, the brush trajectory is planned so as to avoid the weak direction of brush motion when the picture model is composed.

The process of grasping is as shown in Figure 4. First, the position of the handle of a paintbrush is detected using the vision system. The detail of the algorithm is described in the following section because the algorithm used here is based on the algorithm of the tip detection described in the following section. Next, the right hand approaches the paintbrush. When the force sensor in the middle finger detects the contact between the finger and the paintbrush, the system decides that the hand is adequately close to the paintbrush for grasping, and stops the hand motion. Finally, power of the fingers is gradually added. The force on the fingers is watched during this phase, and power is added to the fingers until a firm grasp is realized.

### C. Detecting Tip of Brush

The detection of the tip of a brush is performed using the stereo vision system as Figure 5. First, the position of the handle of a paintbrush is detected, and then the position of the brush's tip is derived from it. In this system, because the paintbrush is grasped by the multi-fingered hand, the relative position of the tip to the arm changes a bit with every trial of grasping. Moreover, the position can change by an unexpected slip of the paintbrush during painting. Therefore, it is required that the tip position is frequently detected in order that the system always knows the correct position. The reason why the tip position is not detected directly is that

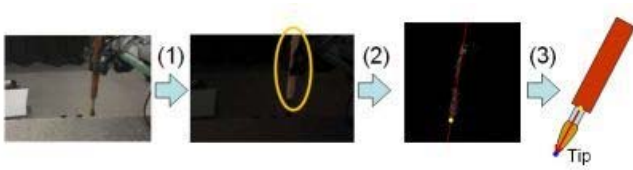


Fig. 5. Detection of the tip of a brush: (1) Selecting pixels belonging to the handle of the paintbrush (2) Calculating the position and orientation of the paintbrush from the position of the pixels in 3D space (3) Obtaining the position of the tip from the position of the handle.

trial	$x$ -axis	$y$ -axis	$z$ -axis
1	0.64	0.89	0.66
2	0.77	1.5	0.80
3	0.90	1.6	0.49

Fig. 6. Evaluation of tip detection [mm]: This table shows the standard deviation of the detected positions.

direct detection is difficult because the brush is deformable and its color changes according to the color of the paint.

First, pixels belonging to the handle are extracted from the captured image (Figure 5-1). The color of the handle is given a parameter here, and the hue and saturation are used for the extraction. The 3D position of each pixel is calculated by stereo calculation. For improving accuracy, the area to detect is limited to the possible area determined from the hand position and rotation. Next, principal component analysis (PCA) is performed on the extracted pixels (in 3D space), and used to calculate the axis of the handle (Figure 5-2). The pixel in the lowest position is regarded as the lowest point of the handle. Finally, the position of the tip is derived from these two factors: the axis and the lower point of the handle (Figure 5-3).

Figure 6 shows the result of measuring the accuracy of tip detection. The tip position was calculated ten times, keeping the paintbrush in the same position. Because the paintbrush does not move, the result must always be same, but some error appears due to the vision system. The table shows the standard deviation of the detected positions. Three trials, in each of which the tip position is detected ten times, are performed in various positions and rotations. In every trial, the standard deviation of the error is within about 1.0 mm. However, although the brush deforms during painting, this algorithm does not consider it. In the actual case, the error of about 10 mm appears by the deformation and it is a bottleneck in the tip detection.

#### D. Detecting Brush-Canvas Contact

Force sensors in the fingers are used for the detection of the contact between a brush and a canvas. The integration of force acting on the three fingers that support a paintbrush ( $F$ ) is focused here. According to Newton's equation, if a paintbrush is regarded as moving in a quasi-static process,  $F$  is formulated as

$$F = -mg + F_{\text{ground}},$$

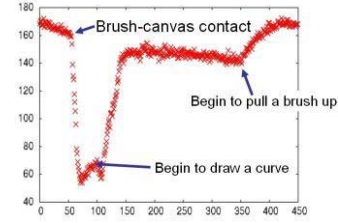


Fig. 7. Force acting on the fingers ( $F$ ): The integration of force acting on the three fingers that support a paintbrush.

where  $m$  is the mass of the paintbrush,  $g$  is the gravity acceleration, and  $F_{\text{ground}}$  is the ground reaction acting on the paintbrush. From this equation, it can be said that  $F$  begins to decrease when the brush contacts a canvas.

Figure 7 shows the change of  $F$  while pushing down a paintbrush, drawing a line, and pulling up the paintbrush. It is seen that  $F$  suddenly decreases when contact of brush and a canvas occurs. Therefore, we set a threshold and detect contact of brush and a canvas.

#### E. Parameterized Paintbrush Technique

Now, the following three paintbrush techniques are parameterized (see Figure 8):

- Leaning a paintbrush to the drawing direction (a lean angle)
- Pushing a paintbrush onto a canvas (an amount of pressure)
- Pulling a paintbrush up gradually (an amount of sweep)

The first parameter is a lean angle of a paintbrush. The drawn lines and curves, especially curves, vary according to the lean angle.

The second parameter is an amount of pressure. When a paintbrush is pushed down upon a canvas, it does not stop moving downward as soon as it contacts the canvas, but it keeps moving for awhile. The duration to keep moving is varied as the parameter. It controls painting pressure.

The third parameter is an amount of sweeping performed by a paintbrush at the end of drawing. In drawing a curve with a paintbrush, the tip motion of the brush tends to lag behind the motion of the arm because the brush deforms during drawing. Therefore, if the arm stops when it reaches its target position, the tip often still remains behind its target position. In order to avoid this situation, a paintbrush is pulled up while gradually moving to the target direction. The parameter determines an amount of this motion.

#### F. Verifying Painted Picture

In painting with a paintbrush, it is difficult to predict how the brush deforms, and so the result of painting is not always the same as the intended picture. In order to cope with this divergence, the system performs a step that is “*checking a result and rewriting it.*” Such a visual feedback step is always performed in human painting, and so it can be considered a basic step in painting.

The implementation is as follows: first, an image of the canvas is captured and color of pixels that are expected to

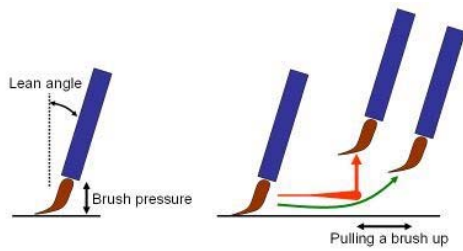


Fig. 8. Parameterized brush technique: leaning a brush to the drawing direction, pushing a brush onto a canvas (left), and pulling a brush up gradually (right).

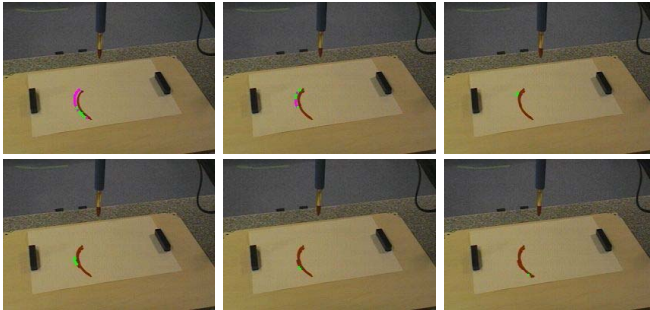


Fig. 9. Visual feedback: The top left is the initial result, and the curve is completed by being gradually modified.

belong to drawn curves is checked. In practice, curves are divided into small line elements, and the check is done for the mean pixel of the elements. The system judges that the curve is correctly painted if the color matches the oil color; otherwise, it judges that the curve is not correctly painted and tries to rewrite the curve. Figure 9 shows the process of the visual feedback. A green point indicates that the pixel is painted correctly, and a purple point indicates that the pixel is not painted correctly. A curve is gradually modified and is finally completed.

In most cases in which a painted result differs from the intended one by deformation of a brush, the result tends to be lesser than the intention. For example, a curve may not reach to the target point or be too flat. Therefore, the incremental method described here can be used to bring a good result in most cases.

### III. COMPOSING A PICTURE MODEL

If we obtain an appropriate 3D model of an object, the robot cannot start painting by directly referring to the 3D model. A bridge between the 3D model and painting is required. For this purpose, we consider a *picture model*, which describes how to represent the target object on a 2D canvas. A picture model is composed by extracting the features of the object from its 3D model and texture. Now, we implement the following two methods for painting.

- Extracting feature lines of an object
- Filling a region with color

These two methods are described below.

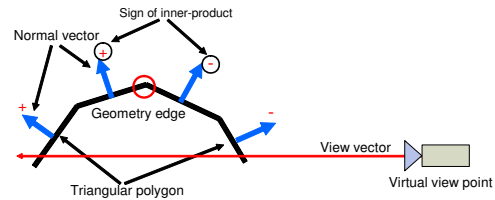


Fig. 10. Geometry edge

#### A. Extracting Feature Lines

One of the typical methods to represent an object is drawing lines that feature the object, such as a silhouette or boundaries of color or pattern. In this paper, we extract geometry edges as the feature lines.

A 3D model consists of triangular patches. A geometry edge is defined as the common edge at which the sign of inner product of the normal vector of a patch and the viewing direction changes (Figure 10). The geometry edges extracted in this way include hidden edges and many very short edges. The hidden edges are eliminated and the short edges are concatenated if they are neighbors. The results of these procedure are shown in Figure 11. In spite of the procedure, geometry edges at this moment are too complex to paint (Figure 12).

Therefore, a simplification algorithm using spline curves is applied. First, end points of edges are clustered, and redundant points are gathered into one point. Next, the new points are connected using C-spline curves. The curves are compared, and close curves are gathered into one curve. Figure 13 shows the simplification.

If we used an XY-plotter to paint a picture, the above procedure of extracting future curves would be sufficient. However, we are trying to paint a picture by a multi-fingered robot, and so an additional procedure is required to consider peculiarity of grasping by the robot. In this system, because grasping is designed to mimic human grasping, it is stable when a brush moves from the left to the right and from the far side to the near side, while it is unstable when a brush moves from the near side to the far side. Therefore, brush trajectory must be designed to avoid such unstable brush motion. The start points of a curve are decided so that a curve is drawn in a more stable way. When an unstable trajectory appears even if the edge point of a curve is chosen as the starting point, the curve is cut into two curves to avoid an unstable brush motion (Figure 14).

#### B. Filling Regions with Color

Drawing feature curves is not sufficient for producing a painting system: filling a region with color is also necessary. In this subsection, the picture model for filling a region is described.

Input of this method is a texture image, rather than a 3D shape. First, segmentation is applied to the input image. In each segment, it is decided how to fill the segment. The gradient at every pixel in a segment is calculated, and a histogram of the pixels is constructed. The peak of

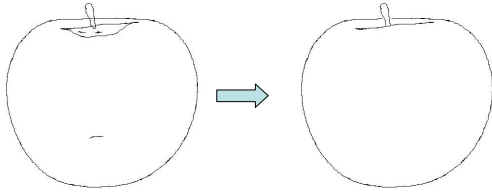


Fig. 11. Procedure of geometry edge extraction: The left figure shows a silhouette before the elimination of hidden edges and very short edges, and the right figure shows it after the procedure.

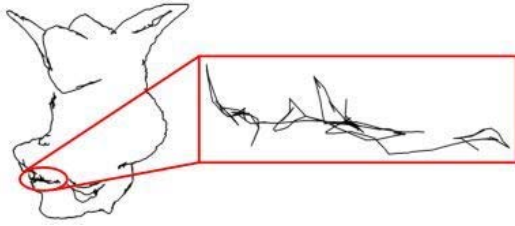


Fig. 12. Complicated feature lines: Feature curves without simplification are very complicated when they are zoomed in.

the histogram indicates the direction of the brush for the corresponding segment.

This phase is performed in hierarchical way. The result is detailed iteratively by changing a brush to a smaller one at every iteration.

The simulation result of the method is shown in Figure 15. The effectiveness of the filling algorithm is especially shown in lady's hair in the middle picture.

#### IV. EXPERIMENT

Figure 16 and 17 are the result of painting by the robot using the method described above.

Figure 16 is the result of painting an apple. The 3D model is obtained from multi-view range images. We use a laser range sensor named Vivid for measuring a small range (a few meters). An adequate number of local images, which overlap each other, are captured from several viewpoints, and they are aligned and merged using an algorithm developed in our laboratory. The top left of Figure 16 is the 3D model. Extracting a silhouette of the 3D model, the picture model is generated. The top middle of Figure 16 is the picture model. The results drawn by the robot are the reminder figures in Figure 16. The results are different for every trial because of the error included in the detecting position of a brush by the vision system.

Figure 17 is the result of painting a human silhouette. In this case, the 3D model is obtained from multi-view camera images. We use video cameras that are arranged to surround an object, and the measurement is performed simultaneously. Using a volume-intersection method [12], the 3D model is calculated from multi-view camera images. The left of Figure 17 shows the 3D model. Subsequent procedures for painting the object are the same as the above case. The

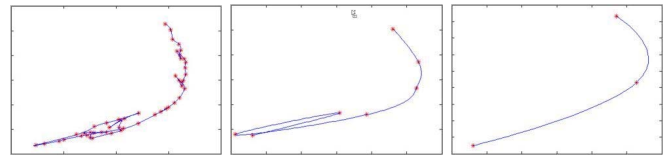


Fig. 13. Simplification with spline curves: The left figure shows original complicated edges, the middle figure shows the result of clustering vertices, and the right figure shows the result of gathering close curves.

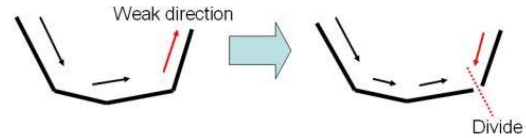


Fig. 14. Planning of brush trajectory: a curve is divided so that the brush trajectory does not include the weak direction.

middle of Figure 17 is the picture model, and the right is the result drawn by the robot.

Because a method to fill a region by the robot is not developed yet although a picture model for filling is already developed, these results are composed only by feature curves such as silhouettes. The filling by the robot is a future project.

#### V. CONCLUSION

In this paper, we developed a painting robot. This system observes an object, obtains the 3D model, generates the picture model extracting features of the model, and paints it on a canvas with a paintbrush. It is an experiment to reproduce the entire process of human painting by a robot system. The features of this system are reconstruction of a 3D model by a multi-camera system, manipulation of a paintbrush with a multi-fingered hand, and modification of a picture by a visual feedback system. In this paper, an apple is selected as a motif and experiments to draw it by a robot are performed.

For our future work, our first priority is *filling a region*. As mentioned above, we have already developed a method to generate a picture model for filling. In order to have it performed by a robot with a multi-fingered hand, We need to introduce an additional operation, such as changing paintbrushes and dipping a paintbrush into water color. Another project is improving a picture model. In this paper, we considered only feature curves and silhouettes, but we are trying to develop more advanced techniques, such as putting appearances from several viewpoints together into one picture like paintings of cubists. Through this study, we believe that we can understand the meaning of *painting*, which is one of the primary behaviors for humans, using analysis-by-synthesis.

#### ACKNOWLEDGMENTS

This work is supported by the Japan Science and Technology Corporation (JST) under the CREST project.



Fig. 15. Filling a region: The upper images are the original picture, and the lower images are the simulation result.

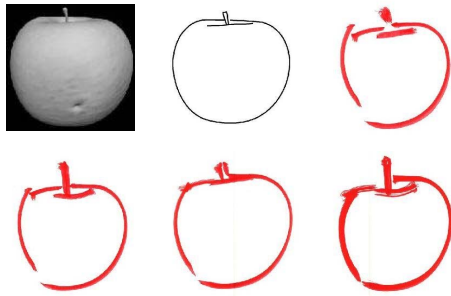


Fig. 16. Painting an apple: These are results of painting an apple. The 3D model is obtained using multi-view range images. The top left is the 3D model, the top middle is the picture model, and the rest are the results drawn by the robot.

## REFERENCES

- [1] Kazuhiko Yokoyama, Hiroyuki Handa, Takakatsu Isozumi, Yutaro Fukase, Kenji Kaneko, Fumio Kanehiro, Yoshihiro Kawai, Fumiaki Tomita, and Hirohisa Hirukawa. Cooperative works by a human and a humanoid robot. In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2985–2991, 2003.
- [2] Hitoshi Hasunuma, Masami Kobayashi, Hisashi Moriyama, Yoshiyuki Itoko, Yoshitaka Yanagihara, Takao Ueno, Kazuhisa Ohya, and Kazuhito Yokoi. A tele-operated humanoid robot drives a lift truck. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2246–2252, 2002.
- [3] Hitoshi Hasunuma, Katsumi Nakashima, Masami Kobayashi, Fumisato Mifune, Yoshitaka Yanagihara, Takao Ueno, Kazuhisa Ohya,

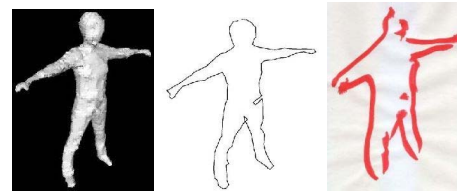


Fig. 17. Painting a human: These are results of painting a human. The 3D model is obtained using multi-view camera images. The left is the 3D model, the middle is the picture model, and the right is the result drawn by the robot.

- and Kazuhito Yokoi. A tele-operated humanoid robot drives a backhoe. In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2998–3004, 2003.
- [4] Shuji Kajita, Humio Kanehiro, Kenji Kaneko, Kiyoshi Fujiwara, Kazuhito Yokoi, and Hirohisa Hirukawa. A realtime pattern generator for biped walking. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA)*, pages 31–37, 2002.
- [5] Koichi Nishiwaki, Satoshi Kagami, Yasuo Kuniyoshi, Masayuki Inaba, and Hirochika Inoue. Online generation of humanoid walking motion based on a fast generation method of motion pattern that follows desired zmp. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2684–2689, 2002.
- [6] Tomomichi Sugihara, Yoshihiko Nakamura, and Hirochika Inoue. Realtime humanoid motion generation through zmp manipulation based on inverted pendulum control. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1404–1409, 2002.
- [7] J. Napier. The prehensile movements of the human hand. *Journal of Bone and Joint Surgery*, 38B(4):902–913, 1956.
- [8] M. R. Cutkosky. On grasp choice, grasp models, and the design of hands for manufacturing tasks. *IEEE Transactions on Robotics and Automation*, 5(3):269–279, 1989.
- [9] S. B. Kang and K. Ikeuchi. Toward automatic robot instruction from perception — mapping human grasps to manipulator grasps. *IEEE Transactions on Robotics and Automation*, 13(1):81–95, 1997.
- [10] Takahiro Yokoyama, Shigeki Kudomi, and Teruaki Mitsui. Developing and implementation of sensing method for novel sophisticated robot teaching — facilitate the reproduction of experienced painter's art —. Technical report, Gifu Prefectural Research Institute of Manufacturing Information Technology, 2003. (in Japanese).
- [11] Koichi Ogawara, Xiaolu Li, and Katsushi Ikeuchi. Human motion tracking by articulated deformable model. In *Proceedings of Meeting on Image Recognition and Understanding (MIRU)*, 2006.
- [12] A. Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 16(2):150–162, 1994.