Machine Vision and Applications (2005) 16: 116–121
Digital Object Identifier (DOI) 10.1007/s00138-004-0163-4

Machine Vision and Applications

Type classification, color estimation, and specific target detection of moving targets on public streets

Osamu Hasegawa[1,2], **Takeo Kanade**[3]

[1] Imaging Science and Engineering Laboratory, Tokyo Institute of Technology, Tokyo, Japan
[2] PRESTO, Japan Science and Technology Agency, Tokyo, Japan
[3] The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA

Received: 11 March 2004 / Accepted: 11 August 2004
Published online: 20 December 2004 –

Abstract. This paper describes a vision system that recognizes moving targets such as vehicles and pedestrians on public streets. This system can: (1) classify targets {vehicle, pedestrian, others} and, for "vehicles," discriminate vehicle types and (2) estimate the main colors of targets. According to the input images to the system, the categories of targets are set as {mule (golf cart for workers), sedan, van, truck, pedestrian (single or plural), and other (such as noise)}. Their colors are set as six color groups {red, orange, yellow; green; blue, light blue; white, silver, gray; dark blue, dark gray, black; dark red, dark orange}. In this experiment, we collected images of targets from 9:00 a.m. to 5:00 p.m. on sunny and cloudy days as system training samples. The recognition ratio was 91.1% under the condition that both the recognition results of type and color agreed with the operator's judgment. In addition, the system can detect predefined specific targets such as delivery vans, post office vans, and police cars by combining recognition results for type and color. The recognition ratio for specific targets was 92.9%. For the classification and estimation of targets, we employed a statistical linear discrimination method (linear discriminant analysis, LDA) and a nonlinear decision rule (weighted K-nearest neighbor rule, K-NN).

1 Introduction

Recently, along with increasing demand for security, a growing number of monitoring cameras has been appearing on public streets. This paper presents a vision system that can recognize moving targets such as vehicles and pedestrians on public streets.[1] The targets of this system are moving outdoors. For this reason, the information gained from the targets is influenced by several factors:

1. Shadows cast by other objects
2. Direction of sunlight
3. Reflection of other objects on the targets
4. Obscuring of targets by other objects

Correspondence to: O. Hasegawa
(e-mail: hasegawa@isl.titech.ac.jp)

[1] In this study, pedestrians and vehicles pass on the street in both directions. Furthermore, the road has no median strip.

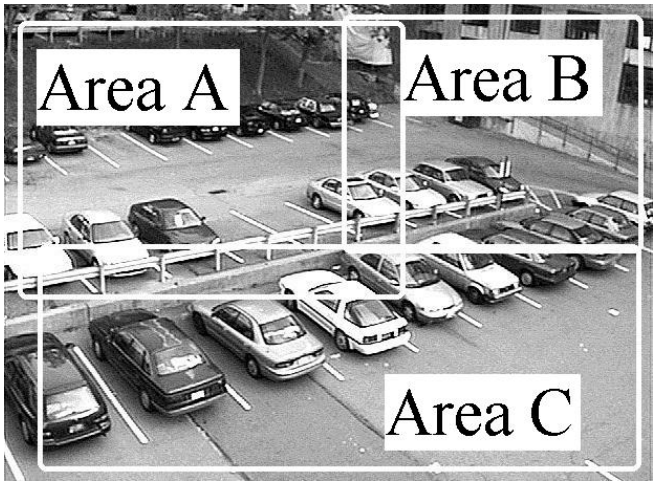

Fig. 1. Input image to the system and its segmentation

Therefore, it is not always possible to obtain valid information simultaneously for both category classification and target color estimation. For example, the shapes of targets could be broken or the color of targets could be affected by strong sunlight reflection or shadows of other objects. To resolve these problems, we present a scheme that can determine the tracks of targets and at the same time independently classify the categories of targets as well as estimate their colors. Moreover, by comparing each result throughout the tracking sequence, the scheme can automatically choose discriminant results with the largest matching probability as the final decision. Figure 1 shows a sample input image used in this system.

Among previous research efforts, many reports [1–4] have concentrated on the search for recognition systems that are suitable for the lanes of a highway to check moving vehicles running in one direction and, for toll stations, to check stopped vehicles. However, no studies have sought to classify and categorize moving targets and estimate their colors and consequently identify specific vehicles on general thoroughfares.

Trucks

Vans

Sedans

Fig. 2. Sample images of targets extracted from area A

Limin et al. [5] expressed the 3D contour of an automobile using 29 structure parameters to estimate a target's 3D orientation and to distinguish the type of automobile from its 2D image. They reported their primary results. Wu et al. [6] also used 29 structural parameters to express the 3D geometry of an automobile, train a neural network, and dinstinguish six types of automobiles from their 2D images with 91% accuracy. These procedures seem robust against the variation of vehicle orientation. However, neither study considered a person to be a target of distinction. In addition, no color information was used. For this reason, it is hard to distinguish specific vehicles such as police cars with color information combined.

Gupte et al. [7] created a system in which the vehicle image obtained from the top view of highway lanes is altered through adaptive background subtraction. The vehicle class is later identified by its width and height. Two classes are identified: cars, such as sedans, and noncars, such as trailers. No color information is used in this study.

2 Outline of this system

The system comprises a fixed camera that is set on the roof of a two-story building and a personal computer (CPU: Pentium III, 400 MHz), which processes the input image from the camera. We use no special hardware for image processing. The camera is a standard CCD camera. The NTSC video signal is input to the system; its input image size is 640×480 pixels.

The operating system (OS) of the computer is Linux; the programming language is C. The system consists of four modules: (a) a module based on background subtraction to segment the target (vehicle or pedestrians) from the image and track it, (b) a module to classify the category of the segmented target, (c) a module to estimate the color of the target, and (d) a module to decide the result from the above category classification and color estimation. Module (a) above, which was studied and developed independently by Lipton et al. [8], can update the background of images according to sunshine conditions or weather diversification, segment the target, assign an ID to each target, and track the targets.

3 Target category recognition

3.1 Outline

This study uses vehicles and persons as targets for recognition. Experimental image collection revealed that automobiles of the same class (such as sedans) differ in shape according to their respective manufacturers, as shown in Fig. 2. Another observation is that two or more persons walk in various relative positional relationships simultaneously.

Generally, the procedures for object distinction are classified roughly into model-based and appearance-based procedures. The former procedure requires construction of a model for each class to which the target belongs. However, we considered it difficult to establish a geometric model for each class that can address all the targets in each class shown in Fig. 2.

This study introduces an appearance-based procedure to solve the above problems. Target images with shadows or missing parts are learned by the system according to their appearance, thereby improving the rate of distinction of such images. Furthermore, another category, "other," is prepared in the category of processing targets. Thus the appearance of targets not amenable to classification in the other categories are distinguished collectively and clearly. This addition further reduces trimming errors at background subtraction and improves the rate of distinction of overlapped images.

3.2 Collection of image samples for learning

Figure 1 shows a sample input image of the experimental system. In this study, the appearance (direction, size, etc.) of the segmented target varies greatly according to the position at which it is segmented. Therefore, as shown in Fig. 1, the input image is divided into areas A, B, and C. This experiment does not include cars parked in a parking space or leaving the garage.

Sample learning images are collected from online input images or videotape images acquired over the course of 16 h from 9:00 a.m. to sunset under sunny or cloudy conditions. Some 2000 learning samples are collected for the recognition system.

The collected sample images are identified by visual inspection of the operator. A category label of {sedan, van, truck, mule, pedestrian, or other} is attached to each identified image. If the sample image is the specific target (in this paper, we specify the FedEx van, UPS van, and police car as specific targets), the specific target label is attached (in which case there are two labels for one sample).

3.3 Learning process of category

Sample images for learning (background-subtracted images) are smoothed in the height and width directions. For each background-subtracted image there are 11 features that compose the 11-dimensional feature vector of an image. The features are:

- Width and height of box that surrounds the background-subtracted image (two features).
- First, second, and third moments of the background-subtracted image in width and height directions ($2 \times 3 = 6$ features).
- Centroid coordinate (x, y) (two features).
- Area of background-subtracted image (one feature).

Using the labels of each category, the above feature vectors are sorted into classes (categories) that correspond to each area (areas A, B, C). Then LDA is used to analyze these classes. Discriminant spaces for category classification are later constructed for their respective areas. Assume that B represents the variance among all classes and W represents variance within a class. They are given by

$$B = \sum_{k=1}^{J} n_k (\bar{x}_k - \bar{x})(\bar{x}_k - \bar{x})^T , \tag{1}$$

$$W = \sum_{k=1}^{J} \sum_{i=1}^{n_k} (x_{k,i} - \bar{x}_k)(x_{k,i} - \bar{x}_k)^T . \tag{2}$$

The meanings of these parameters are:
J: number of classes
n: number of samples in all classes
n_k: number of samples in class k
$\bar{x}$: mean vector of all vectors
$\bar{x}_k$: mean vector of vectors in class k
$x_{k,i}$: vector of target i belonging to class k
T: denotation of matrix transpose

Next, we solve the eigenvalue problem

$$W^{-1}B \tag{3}$$

to get the eigenvalue λ_i and eigenvector b_i. Here, we assume that the dimension of the input feature vector is N, and

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N . \tag{4}$$

We use the eigenvector b_i as the base of the discriminant space and obtain the contribution ratio of the corresponding eigenvector. In this study, we assume that the dimension of discriminant space has reached its upper limit (which is assumed to be the M dimension) when the cumulative contribution ratio of eigenvector reaches 99.9%:[2]

$$0.999 \times \sum_{i=1}^{N} \lambda_i \quad \leq \sum_{i=1}^{M} \lambda_i . \tag{5}$$

[2] In this study, $N = 11$. When the cumulative contribution ratio of the eigenvector is greater than 99.9%, the dimension of the discriminant space is $M = 10$.

If L represents the final dimensional number of discriminant space, the following relationship exists between M and class number J:

$$L = min(J - 1, M) . \tag{6}$$

Consequently, the formula to translate the input N-dimensional image feature vector into L-dimensional discriminant space is

$$\underset{L \times 1}{y} \quad = \quad \underset{L \times N}{[b_1 b_2 \ldots b_L]^T} \quad \underset{N \times 1}{x} .$$

3.4 Automatic classification of category

All the following classification processes are performed automatically by the system.

1. Assume that an image is extracted from a target with identification number "id" at a time t by background subtraction[3] and that the image is smoothed in the width and height directions.
2. In accordance with Sect. 3.3, an 11-dimensional feature vector ($\vec{x}_s^{(id,t)}$) of the smoothed image is composed (s denotes shape).
3. $\vec{x}_s^{(id,t)}$ is projected onto the discriminant space that corresponds to the area where the image is segmented.
4. Using Euclidean distance, K nearest vectors,[4] $\vec{y}_{s(k,c)}$ (c denotes the category of the vector $\vec{y}_{s(k)}$, and k is the index of the nearest vector), are extracted from 2000 prelearned sample vectors.
5. The inverse of the "distance" between $\vec{x}_s^{(id,t)}$ and $\vec{y}_{s(k,c)}$ is calculated, and the "weight" corresponding to every category is defined as follows:

$$weight(\vec{x}_s^{(id,t)}, S_i) , \tag{7}$$

$$\overset{def}{\longleftrightarrow} \sum_{if\ c=i} \frac{1}{\| \vec{x}_s^{(id,t)} - \vec{y}_{s(k,c)} , \|} \tag{8}$$

where $weight(\vec{x}_s^{(id,t)}, S_i)$ is the weight of category S_i corresponding to $\vec{x}_s^{(id,t)}$.
6. The category corresponding to the maximum weight

$$S^{(id,t)} \leftarrow \max_i (weight(\vec{x}_s^{(id,t)}, S_i)) \tag{9}$$

is the classification result of the shape of $\vec{x}_s^{(id,t)}$. These category results and the corresponding weights are saved:

$$class(id, t) = \{S^{(id,t)}, weight(\vec{x}_s^{(id,t)}, S^{(id,t)})\} . \tag{10}$$

4 Color estimation of target

4.1 Color learning

1. Collect sample color texture images of vehicles or pedestrians shot at various times in sunny (under sunlight or in

[3] The "id" is assigned by the target extraction and tracking module [8].

[4] In the experiment, $K = 10$.

shadow) or cloudy weather, as variously as possible. Divide these samples into two subclasses: sunny or cloudy. Extract single colors from the samples that contain varied colors;[5] operators attach the appropriate label of {red-orange-yellow, green, blue-light blue, white-silver-gray, dark blue-dark green-black, dark red-dark orange} to these sample colors based on their sense.
About 1500 samples are segmented from images of sunny days; about 1000 samples are from images of cloudy days.

2. For each labeled image, extract 25 pixels from the top left corner by extracting every third pixel along the width and height directions. Transform these data from the (R, G, B) color space into the $(I1, I2, I3)$ color space and then average them.[6] Here, the feature vector of each color sample is a 3D vector:

$$I1 = (R + G + B)/3.0\,, \tag{11}$$

$$I2 = (R - B)/2.0\,, \tag{12}$$

$$I3 = (2.0 \times G - R - B)/4.0\,. \tag{13}$$

3. Analyze the vector of samples from sunny days or cloudy days using LDA. Form two discriminant spaces using color estimation for sunny days or cloudy days.
4. Estimate the color of all pixels of the specific target's image using corresponding discriminant space according to weather conditions; then attach the best-matching color label (nearest neighbor) to them. Count the pixels with the same color label and save the ratio between color labels in a color database.

4.2 Automatic color classification

The following processes are performed automatically by the system.

1. Presume that an image is extracted from a target with an identification number "id"[7] at time "t" by background subtraction and extract the color texture image of this "id" from the input image.
2. Transform all extracted color values from the (R, G, B) color space into the $(I1, I2, I3)$ color space [9] and compose a 3D feature vector group $\vec{x}_c^{(id,t,n)}$. Here, "c" denotes color and "n" is the index of transformed data.
3. Project the $\vec{x}_c^{(id,t,n)}$ into the discriminant space corresponding to the weather and then extract K sets of nearest vectors $\vec{y}_{c(n,k,c)}$ from the prelearned [1500 (sunny) or 1000 (cloudy)] sample vectors. Here, in $\vec{y}_{c(n,k,c)}$, "k" is the index of nearest vectors and "c" is the color group (class) that $\vec{y}_{c(n,k,c)}$ belongs to.
4. Calculate the inverse of the "distance" between $\vec{x}_c^{(id,t,n)}$ and $\vec{y}_{c(n,k,c)}$. Then define the "weight" corresponding to each color group as follows:

$$weight(\vec{x}_c^{(id,t,n)}, C_i)\,, \tag{14}$$

$$\stackrel{def}{\longleftrightarrow} \sum_{if\ c=i} \frac{1}{\parallel \vec{x}_c^{(id,t,n)} - \vec{y}_{c(n,k,c)} \parallel}\,. \tag{15}$$

Here, $weight(\vec{x}_c^{(id,t,n)}, C_i)$ is the weight of color group C_i corresponding to $\vec{x}_c^{(id,t,n)}$.

5. Perform the above process for all pixels. Calculate the color group "weight" according to the "weight" of each pixel and estimate the matched color group based on the maximum "weight":

$$C^{(id,t)} \leftarrow \max_i(weight(\vec{x}_c^{(id,t)}, C_i))\,. \tag{16}$$

Save the matched color group and "weight":

$$color(id, t)\,, \tag{17}$$

$$= \{C^{(id,t)}, weight(\vec{x}_c^{(id,t)}, C^{(id,t)})\}\,. \tag{18}$$

5 Final decision

Processes that were described before Sect. 4 are conducted for the segmented tracked image. The module that yields a final decision synthesizes the above process results and then produces a final determination.

As mentioned in Sect. 2, when a new target appears in the input image, the tracking module extracts it and assigns an ID number to it while tracking its movement. The module that produces the final decision calculates the maximum "weight" to determine the most matched shape category and color group independently and outputs the "final output" x^{id}:

$$weight(\vec{x}_s^{(id,t')}, S^{(id,t')})\,, \tag{19}$$

$$= \max_t(weight(\vec{x}_s^{(id,t)}, S^{(id,t)}))\,, \tag{20}$$

$$weight(\vec{x}_c^{(id,t')}, C^{(id,t')})\,, \tag{21}$$

$$= \max_t(weight(\vec{x}_c^{(id,t)}, C^{(id,t)}))\,, \tag{22}$$

$$S^{id} = S^{(id,t')} \ \leftarrow\ class(id, t')\,, \tag{23}$$

$$C^{id} = C^{(id,t')} \ \leftarrow\ color(id, t')\,, \tag{24}$$

$$x^{id} = \{S^{id}, C^{id}\}\,. \tag{25}$$

For the top K sets of learning vectors during category classification mentioned in Sect. 3.4.4, the "weight" of vectors with specific target labels is counted for each specific target. Using the scheme in Sect. 4.2.4, calculate the ratio between the weight $weight(\vec{x}_c^{(id,t)}, C_i)$ of each color group.

Here, for all specific targets whose counted weight is larger than a predefined threshold, calculate the mean square error, "$error$", of the ratio of the target with the largest counted weight according to its color database "$db^{(id,t)}$." If the "$error$" is less than an error tolerance,[8] this target $x^{(id,t)}$ will be classified as a specific target $Sp^{(id,t)}$.

If the "$error$" is larger than the error tolerance, go through the same process for the specific target with the second largest counted weight and obtain a new "$error$". If this "$error$" is less

[5] For example, the color of the body of the FedEx van is white; the painted color of the FedEx logo is red and blue.

[6] Experimental results show that the $(I1, I2, I3)$ color space achieves better classification results than the (R, G, B) [9] or the (Y, I, Q) color space.

[7] The "id" is assigned by the target-extraction and tracking module [8] (Sect. 2).

[8] The error tolerance is determined based on experimentation.

than the error tolerance, the target will be judged as a specific target. If this "$error$" is larger than the error tolerance, or if there is no target with the second largest counted weight, the processing target will be judged to be a nonspecific target.

If $Sp^{(id,t)}$ is obtained, save this result and the "$error$":

$$specific(id,t) = \{Sp^{(id,t)}, error(\vec{x}^{(id,t)}, db^{(id,t)})\}\,; \quad (26)$$

then, throughout the tracking sequence, find the specific target with the minimum "$error$." This specific target will be the last result Sp^{id}:

$$error(\vec{x}^{(id,t')}, db^{(id,t')})\,, \quad (27)$$

$$= \min_{t}(error(\vec{x}^{(id,t)}, db^{(id,t)}))\,, \quad (28)$$

$$Sp^{id} = Sp^{(id,t')} \;\leftarrow\; specific(id,t')\,. \quad (29)$$

6 Experimental results and discussion

6.1 Sample of processing display

Figure 3 shows a sample of an input image and classified results. The upper left part of the box that surrounds each target shows the results: (1) The possibility of detecting specific targets $Sp^{(id,t)}$, (2) the color group $C^{(id,t)}$, and (3) the type of vehicle $S^{(id,t)}$. Here, the estimation results of colors are shown with their associated symbols:

- Rd, Or, Yl: red, orange, yellow
- Gr: green
- Bl, LBl: blue, light blue
- Wh, Sl, Gy: white, silver, gray
- DBl, DGy, Bl: dark blue, dark gray, black
- DRd, DOr: dark red, dark orange

To assist the operator's understanding, font colors of the classified results shown in the display are automatically determined by the recognition reliability as follows: "(high reliability) red $>$ green $>$ white $>$ not displayed (low reliability)" (we substituted white for red and gray for green for these font colors in Figs. 3 and 4). The threshold of font color change is determined using the experimental results.

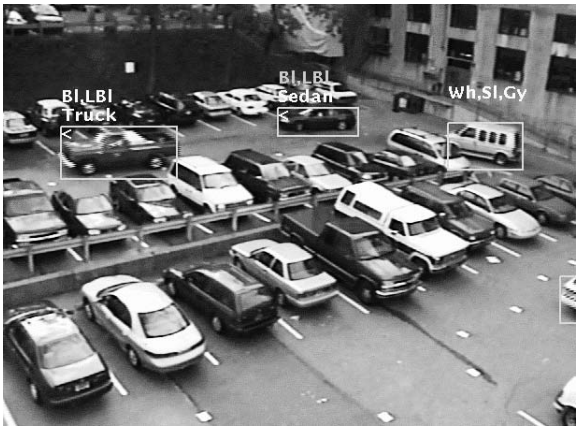

Fig. 3. System output example

a FedEx van **b** Taxi

Fig. 4. Detection of a specific target by type and color. **a** Predefined target. **b** Nonpredefined target (yellow sedan: taxi)

Table 2. Results of specific target detection

	Detection rate	False alarms
FedEx van	11/12 (91.0%)	3/88 (3.4%)
Mail van	12/13 (92.3%)	7/87 (8.0%)
Police car	16/17 (94.1%)	12/83 (14.5%)
Average	*92.9%*	*8.5%*

The processing speed at classification (including target segmentation, shape classification, color estimation, and specific target specification) is two or three cycles per second where the input image contains only one target to process (CPU is Pentium III, 400 MHz).

6.2 Experimental results

The experiments are conducted with the following procedures. Considering both the solar location and weather, images from the camera are input to the system online between 9:00 a.m. and 10:00 a.m., 1:00 p.m. and 2:00 p.m., or 4:00 p.m. and 5:00 p.m. on sunny or cloudy days. No input images are included in the sample images for learning. The number of targets to process that appear in the images amounts to 180.

In this experiment, the results for the final category and the color decision by the system are counted as a correct answer only when both classification results match results of human sight judgment. The average of correct answer rates approaches 91.1% (Table 1).

Classification results on cloudy days are better than those for sunny and cloudy days because the images on cloudy days are more stable. Regarding the influence of the target shadow, the shadow seen in part of Fig. 2 is included as part of the target in the learning data.

Table 2 shows experimental results of the detection of specific targets. Figure 4a shows an example in which one specific target FedEx van is detected. Additionally in this test, the functions of category classification and color estimation are applied to detection of specific targets that have not been learned in advance. Taxis in the neighborhood of the test site are all yellow sedans. Therefore, using the results of category classification and color estimation, we compel the system to output red-orange-yellow sedans as possible taxis. Consequently, the detection rate became 95.2%. Figure 4b shows one detected taxi example using this method.

Table 1. Experimental results

	Pedestrian	Sedan	Van	Truck	Mule	Other	Total	Correct	%
Pedestrian	67	0	0	0	0	7	74	67	90.5%
Sedan	0	33	2	0	0	0	35	33	94.2%
Van	0	1	24	0	0	0	25	24	96.0%
Truck	0	2	1	12	0	0	15	12	80.0%
Mule	0	0	0	0	15	1	16	15	93.8%
Other	0	2	0	0	0	13	15	13	86.7%
Total							180	164	*91.1*%

7 Conclusion

This paper described a vision system for category classification and color estimation from video images of a street. The proposed method is also applicable to the detection of specific targets.

Acknowledgements. We would like to thank the members of CMU VSAM Project.

References

1. Koller D, Weber J, Huang T, Malik J, Ogasawara G, Rao B, Russel S (1994) Towards robust automatic traffic scene analysis in real-time. In: Proc. international conference on pattern recognition, pp 126–131
2. Beymer D, McLauchlan P, Coifman B, Malik J (1997) A real-time computer vision system for measuring traffic parameters. In: Proc. international conference on computer vision and pattern recognition, pp 495–501
3. Mantri S, Bullock D, Garret J, Jr (1997) Vehicle detection using a hardware-implemented neural net. IEEE Expert 12(1):15–21
4. Ikeda T, Ohnaka S, Mizoguchi M (1996) Traffic measurement with a roadside vision system – individual tracking of overlapped vehicles. In: Proc. international conference on pattern recognition, pp 859–864
5. Xia Limin (2002) Vehicle shape recovery and recognition using generic models. In: Proc. 4th world congress on intelligent control and automation, pp 1055–1059
6. Wei Wu, Zhang QiSen, Wang Mingjun (2001) A method of vehicle classification using models and neural networks. In: Proc. IEEE 53rd conference on vehicular technology (VTC2001), 4:3022–3026
7. Gupte S, Masoud O, Martin RFK, Papanikolopoulos NP (2002) Detection and classification of vehicles. IEEE Trans Intell Transport Syst 3:37–47
8. Lipton A, Fujiyoshi H, Patil R (1998) Moving target classification and tracking from real-time video. In: Proc. workshop on application of computer vision, pp 8–14
9. Ohta Y, Kanade T, Sakai T (1980) Color information for region segmentation. Comput Graph Image Process 13:222–241

Osamu Hasegawa received his Dr.Eng. degree in electronic engineering from the University of Tokyo in 1993. He was a research scientist at Electrotechnical Lab (ETL) from 1993 to 1999 and at Advanced Industrial Science and Technology (AIST) from 2000 to 2002. From 1999 to 2000 he was a visiting scientist at the Robotics Institute, Carnegie Mellon University. In 2002 he became a faculty member of the Imaging Science and Engineering Lab, Tokyo Institute of Technology. In 2002, he was jointly appointed researcher at PRESTO, Japan Science and Technology Agency (JST). He is a member of the IEEE Computer Society, IEICE, and IPSJ.

Takeo Kanade received his Ph.D. in electrical engineering from Kyoto University, Japan, in 1974. After being on the faculty of the Department of Information Science, Kyoto University, he joined the Computer Science Department and Robotics Institute in 1980. He became an associate professor in 1982, full professor in 1985, the U.A. and Helen Whitaker Professor in 1993, and university professor in 1998. He was the director of the Robotics Institute from 1992 to spring 2001 and served as the founding chairman (1989–1993) of the robotics Ph.D. program at CMU, probably the first of its kind in the world. He has worked in many areas of robotics including manipulators, sensors, computer vision, multimedia applications, and autonomous robots, and has authored or coauthored more than 200 papers on these topics. He was founding editor of the International Journal of Computer Vision. His professional honors include: election to the National Academy of Engineering, a fellow of IEEE, a fellow of ACM, a fellow of the American Association of Artificial Intelligence; recipient of several awards including the C & C Award, the Joseph Engelberger Award, the Yokogawa Prize, the JARA Award, the Otto Franc Award, and the Marr Prize Award.