Paper:

Visual Attention Region Prediction Based on Eye Tracking Using Fuzzy Inference

Mao Wang*, Yoichiro Maeda, and Yasutake Takahashi*****

*Department of System Design Engineering, Graduate School of Engineering, University of Fukui
3-9-1 Bunkyo, Fukui 910-8507, Japan
E-mail: mwang@ir.his.u-fukui.ac.jp
**Department of Robotics, Faculty of Engineering, Osaka Institute of Technology
5-16-1 Omiya, Asahi-ku, Osaka 535-8585, Japan
E-mail: maeda@bme.oit.ac.jp
***Department of Human and Artificial Intelligent Systems, Graduate School of Engineering, University of Fukui
3-9-1 Bunkyo, Fukui 910-8507, Japan
E-mail: yasutake@ir.his.u-fukui.ac.jp
[Received November 2, 2013; accepted April 12, 2014]

Visual attention region prediction has attracted the attention of intelligent systems researchers because it makes the interaction between human beings and intelligent nonhuman agents to be more intelligent. Visual attention region prediction uses multiple input factors such as gestures, face images and eye gaze position. Physically, disabled persons may find it difficult to move in some way. In this paper, we propose using gaze position estimation as input to a prediction system achieved by extracting image features. Our approach is divided into two parts: user gaze estimation and visual attention region inference. The neural network has been used in user gaze estimation as the decision making unit, following which the user gaze position at the computer screen is then estimated. We proposed that prediction in visual attention region inference of the visual attention region be inferred by using fuzzy inference after image feature maps and saliency maps have been extracted and computed. User experiments conducted to evaluate the prediction accuracy of our proposed method surveyed prediction results. These results indicated that the prediction we proposed performs better at the attention regions position prediction level depending on the image.

Keywords: visual attention, eye tracking, neural network, saliency map, fuzzy inference

1. Introduction

An investigation by the Ministry of Health, Labour and Welfare of Japan in 2011 indicated that about 3.94 million persons in Japan's population are physically challenged. This means that they face functioning problems in daily life. Researchers and research groups now focus on research related to solve these daily problems, such as in communication and mobility, to improve quality of life, the ability to live independently, and how to integrate themselves better in society. Government departments have also supported this work.

Developments in assistive technology have progressed and promising strides have been made in communication between human users and machines to the point where the physically challenged express their intentions, get information from the outside and even get their purposes effectively implemented through assistive technology executed by machines and robots. However, advances in communication by human users to machines have been modest using the keyboard, mouse, joystick, and tactile screen, all of which are manual, although automatic ones include voice and gesture.

Eye tracking technology has special advantages in application in area such as amyotrophic lateral sclerosis (ALS) and is useful in aiding interactions with computer and others. Unlike the methods mentioned above, interaction through eye tracking may feel convenient and direct, especially for users needing to interact with computers but unable to manage a keyboard or mouse. For instance, ALS victims, who ultimately lose any ability to initiate and control voluntary movement, are able to use eye movement in eye-tracking-based interaction.

The use of intention recognition, for example, recognizing the intention of a user or agent by analyzing their actions or changes in state, has become important in intelligent systems research. Specifically, intention recognition makes human-computer interaction (HCI) convenient, and many intention recognition approaches have been proposed.

Much of the early work in intention recognition has been in the context of automatic speech understanding and response [1]. Pynadath et al., for example, achieved plan recognition for a traffic monitoring problem by exploiting the context by using a general Bayesian framework [2]. Pereira et al. [3] described an approach to intention recognition that combines dynamically configurable and situation-sensitive causal Bayes networks and plan

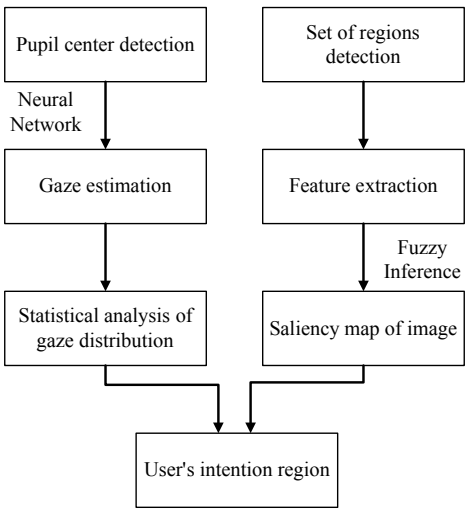


Fig. 1. Overall proposed procedure.

generation [4, 5]. Mao and Gratch present a utility-based approach to solving the recognition of intention that realized by incrementally using plan knowledge and observation to change state probabilities [6].

Probability is a major factor for inferring intention. Another way is to automatically capture bottom-up salient stimuli and to use volitional shifts guided by top-down context factors [7, 8] where bottom-up salient stimuli are the factors external to the user and top-down contexts are internal factors. All of the methods above have a single common denominator: one or more characteristic values of images shown to user have been extracted, calculated, and combined as the basis for inferring user intention.

However, some characteristic of image may be difficult to measure by exact feature values. The grayscale of an image, for example, may indeed be able to be calculated but true feelings in relation to it may be very different depending on the user, which further means that inference or recognition results may also be affected.

This paper focused on visual attention region prediction system inspired on saliency maps based on eye tracking. The objective of this work is to present a novel approach that improves attention prediction performance based on the saliency map using fuzzy inference. Fuzzy inference employing image features as input allows us to combine features and infer very flexibly the intuitive decision rules based on visual perception principles.

2. Overview of Proposed System

The overall procedural flow of the system proposed for visual attention region prediction is summarized in **Fig. 1** and centers on 4 aspects.

First, visual attention region prediction based on eye tracking determines the real-time user's gaze position accurately. Previous research has revealed that eye tracking is divided into two modes: that based on a remote camera and that based on a wearable device [9]. Both used in commercial fields are very expensive, considering their popularization, so we design eye tracking that is low-cost, easy to use, and highly accuracy.

Second, we propose an approach based on image processing and using fuzzy theory to infer a saliency map [7] of an image in user's scene. According to [7], a saliency map is obtained by summing up a few feature maps of an image. The method here, however, has the weakness of ignoring feature importance in the decision process of saliency map. Unlike the method used mentioned above in [7], we get a saliency map by using fuzzy inference based on the characteristic of the regions. Fuzzy rules are made based on their importance, which avoids facing the problem that feature importance cannot be reflected.

Third, because characteristics in an image reveal only that regions underwent image processing and the fuzzy inference above is easily paid attention to compared to the rest, the user's region of attention cannot be inferred or decided based only on image characteristics. So we determine the user's gaze distribution in the scene in real time by analyzing gaze position data in a period decided beforehand to verify the results obtained.

Fourth, the image saliency map is compared to the analyzed gaze distribution results to get reliable results for the user's attention region.

3. Eye Tracking Framework

We now introduce the proposed eye tracking device and the method used for calculating the center of the pupil and analyzing gaze distribution.

3.1. Proposed Eye Tracking Device

Eye tracking technology is divided into two modes: remote camera based mode and wearable device based mode.

The remote camera-based mode has the advantage of being non-intrusive, convenient and applicable to different computer applications. It has the disadvantages however, of requiring a high-resolution camera due to how far the camera is from the user's eye and more than one camera pan-tilt device must be need. All of this increases system complexity and cost. The wearable device based mode estimates user's gaze through a camera with near infrared (NIR) light illuminators attached to a glass frame or a helmet. These eye tracking systems developed for commercial use are based on the remote camera such as Tobii TX300 eye tracker manufactured by Tobii technology. These eye tracking systems have the disadvantages of being very expensive. So development of a cheaper and simple eye tracking system is needed.

As shown in **Fig. 2** and detailed in **Table 1**, the eye tracking hardware we fabricated is wearable device including an eye capture camera attached with NIR LED. To make eye tracking easier, we illuminate the eye with IR light and observe it through an IR sensitive camera with a visible light filter. After doing this the iris of the

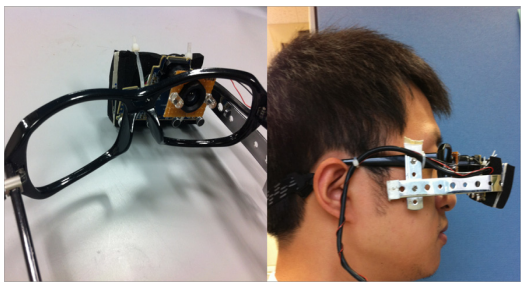


Fig. 2. Proposed eye tracking prototype.

Table 1. Specifications of proposed prototype.

CCD camera	Spatial resolution	640×480 pixels
	Frame rate	60 fps
	Lens focus	fixed
NIR LED	Wave length	940 nm
	Luminous intensity	40 mW/Sr
	Angle	5°

eye turns completely white and the pupil standing out as a high-contrast black dot. It's worth mentioning that the total price of the eye tracking system is about 6800 JPY (68 USD). Among commercially, available eye tracking system, for example, the View Tracker proposed by DITECT Corporation, similar to ours, costs about 1,500,000 JPY (15,000 USD).

3.2. Proposed Eye Tracking Software

This section covers pupil center detection and gaze estimation as shown in **Fig. 1**.

3.2.1. Pupil Center Detection

Pupil center detection is the first part of an eye tracking system, while being the most important part at the same time [4]. We detect the pupil center through the eye image processing shown in **Fig. 3**.

We first capture the eye image in step 1 using a CCD camera and process the binary image, as shown in **Fig. 4**. In step 2, image contrast is increased to make detection process easier. Although mathematical transformation is a large change, it is generally not apparent in the image. The program searches for any blobs existing in the image and records the feature points of the optimal one after filtering in step 3. In step 4, utilizing a Sklansky algorithm [10], the convex shape of feature points is calculated. Finally, pupil center coordinates are obtained by calculating the geometric center after elliptical fitting in step 5. Pupil center coordinates are obtained as shown in ④ of **Fig. 4**.

3.2.2. Gaze Estimation

The primary task in eye tracking system is estimating user's gaze, which is also the foundation of interaction between the human user and the computer in this method.

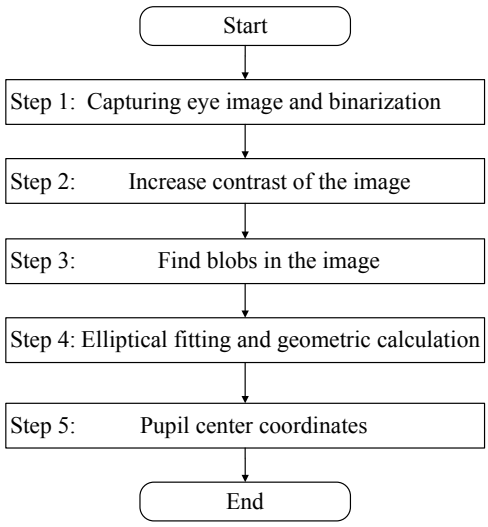


Fig. 3. Pupil center detection algorithm flow.

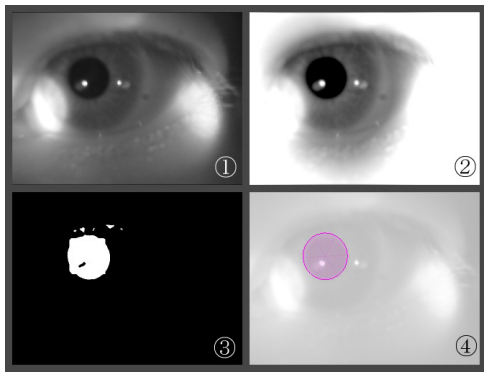


Fig. 4. Eye image capture and pupil center detection.

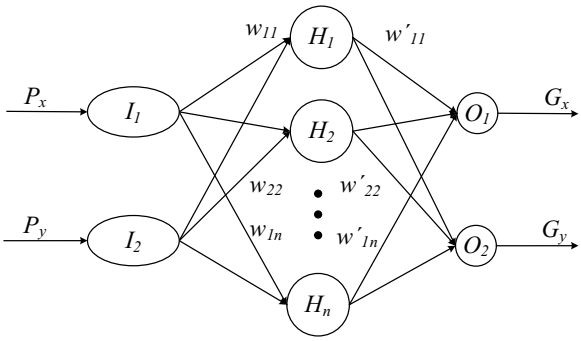


Fig. 5. Neural network for gaze estimating.

Gaze estimation is achieved using a neural network to improve system robustness and adaptability. The calibration process based on pupil center coordinates obtained in Section 3.2.1, uses the two-input and two-output neural network with a standard back propagation algorithm as shown in **Fig. 5**.

Where input P_x, P_y and output G_x, G_y are pupil center coordinates on the 2D camera image plane and the user's gaze position coordinates on the computer screen. w_{1i} is the weight between input node (I_1) and hidden node H_i.

Table 2. Neural network parameter setting.

Desired error	$\leq 0.1\%$
Maximum trial number	3000
Number of layers	3
Number of hidden neurons	9
Learning rate	0.7
Input neurons	2
Input neurons	2

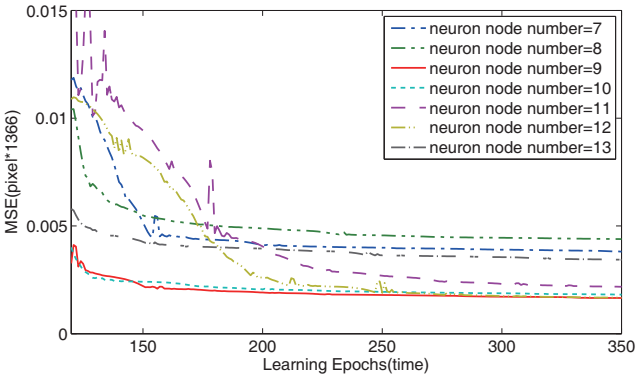


Fig. 6. Training processes of NN according to different hidden neuron nodes number.

We use a sigmoid function as the transmission function. The parameters of NN are shown in **Table 2**.

In **Table 2**, desired error is 0.001 at each pixel. In our program, neural network inputs must in the range of 0 to 1. The data source is actually screen positions that in our experiments are within a range of 0 to 1366 pixels. The max trails number above was set based on experience. **Fig. 6** shows mean square error (MSE) with different hidden neural network neuron number is based on learning trails from neural network training. Note that the training process for neuron number of 9 has fast convergence and a minimum MSE, so we chose 9 as our neural network's hidden neuron number.

Output value G_x of the neural network is calculated as follows:

$$G_x = \frac{1}{1+\exp\left(-\sum_{i=1}^{n}\frac{1}{1+\exp\left(-\sum_{j=1}^{2} I_j w_{ji}\right)} w_{ij}\right)} \quad (1)$$

Output value G_y is calculated by using the same method. Computer screen is 1366×768 pixels, making the range of G_x 0 to 1366 and that of G_y 0 to 768.

In the calibration process, developers usually use defined points such as calibration points [11]. Sometimes this results in mistaken calibration guessing that experiments be repeated. Specifically, users performing an experiment several times move their gaze to the next prospective point before previous calibration finishes. To give calibration greater universality and reduce the possibility of users anticipating subsequent calibration process, our experiment use random calibration, i.e., positions of individual points are given randomly to eliminate user anticipation.

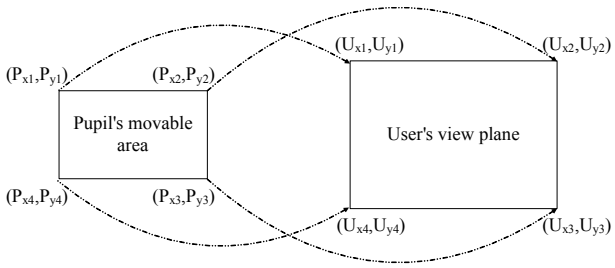


Fig. 7. Coordinate mapping between user view plane and pupil move area.

The set consisting of all points must cover the whole screen, as shown in **Fig. 7**. The rectangle at the right is the user view plane, i.e., the computer screen in experiment. Based on computer screen resolution, in **Fig. 7**, (U_{x1}, U_{y1}) are upper-left corner coordinates $(0,0)$, (U_{x2}, U_{y2}) are for upper-right, (U_{x3}, U_{y3}) are for lower-right, and (U_{x4}, U_{y4}) are for lower-left. Coordinate values are $(1366,0)$, $(1366,768)$, and $(0,768)$. The rectangle at left stands for the pupil movement area. When users look at the upper-left screen corner, where coordinates (U_{x1}, U_{y1}) are the coordinate values (P_{x1}, P_{y1}) are $(190,144)$ in the eye image and resolution is 640×480 pixels. Seen from the other three corners, coordinates (P_{x2}, P_{y2}), (P_{x3}, P_{y3}), and (P_{x4}, P_{y4}) are $(434,148)$, $(440,329)$, and $(183,327)$. In theory, if coordinates (P_{x1}, P_{y1}) and (P_{x3}, P_{y3}) are precise, coordinates (P_{x2}, P_{y2}) and (P_{x4}, P_{y4}) are $(440,144)$ and $(190,329)$. We assume that user's head dose not remain perfectly still in experiments, thus causing slight but acceptable error.

3.3. Gaze Distribution Analysis

As mentioned earlier, the user's visual attention region cannot be decided only on image features, because different users have different interests. These different interests are reflected in gaze distribution, but the attention region cannot be estimated using only gaze position data because such an estimation becomes a direct judgment losing the prediction sense.

For these reasons, we make a mathematical statistics of gaze distribution in a period after user looked at the image at first. Compared to the saliency map based on image processing, as explained in the next section, the user's visual attention regions are decided. In the example of user gaze distribution is in **Fig. 8**, the X-axis is 0 to 1366 and the Y-axis is 0 to 768, corresponding to the screen resolution. The sampling time period is 4 s. **Fig. 9** is the user gaze distribution after normalization.

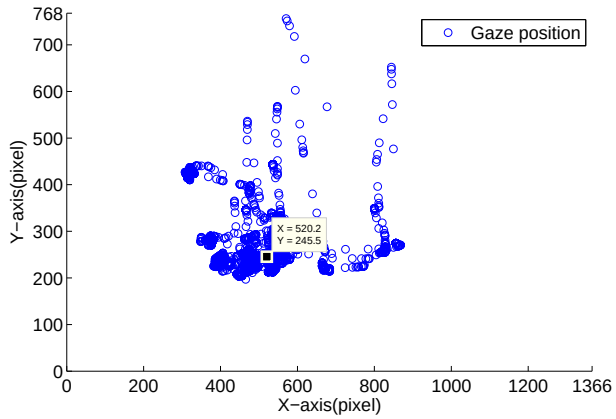


Fig. 8. Distribution of user gaze position.

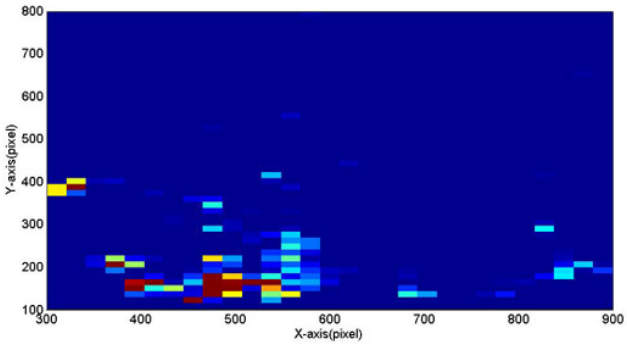


Fig. 9. Statistics of user gaze distribution.

4. Saliency Map by Fuzzy Inference

Most attention models are based on a saliency map and a dynamic process for visiting saliency maxima. Itti et al. [7] introduced a model for bottom-up selective attention based on serially scanning of a saliency map, which is calculated from local feature contrast, for salient locations in the order of decreasing saliency. The saliency map is based entirely on image features and was originally designed to explain converting attention on simple stimuli. Much research done on saliency maps has clarified some image features and combined them by simply summing mathematically [8].

This has its weaknesses however, the saliency map of an image is for example, based on the three feature, i.e., color, intensity, and orientation. Simply summing them gives them the same simultaneous importance. Based on experiments in [12], however few researchers pay equal attention to all of them. The color feature of an image, for example, actually takes more attention from observers than other two features, showing that the method proposed by them may not be very reasonable in some situations. To clarify this issue, we propose calculating saliency map by using fuzzy inference based on the image features so that the importance of all features is reflected in fuzzy rules.

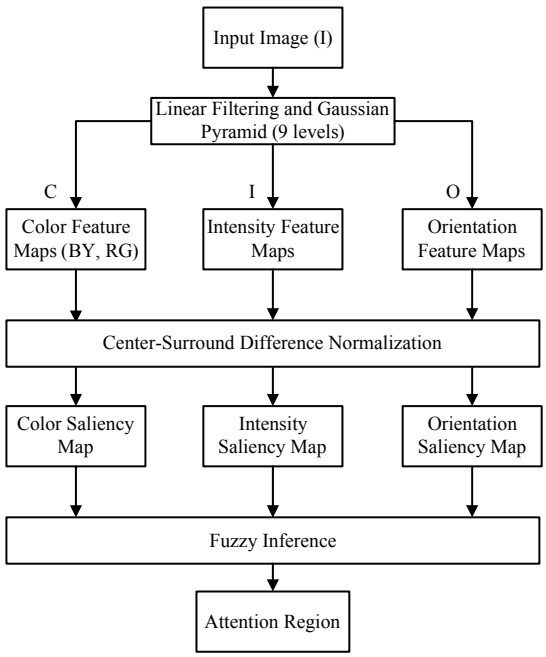


Fig. 10. Architecture of calculating feature map.

4.1. Feature Maps

This section details our calculation framework for building feature saliency maps. For a color input image, we calculate feature maps for color, intensity, and orientation contrast on different scales, as shown in **Fig. 10**.

Linear filtering in **Fig. 10** is used to calculate the center-surround differences of features on 9 scales. The input image is subsampled into a dyadic Gaussian pyramid by convolution with a linearly separable Gaussian filter and extraction by a factor of two [13]. This also means that only half of image pixels are sampled and that processing speed is reduced by half. Gaussian pyramids are used to calculate the center-surround differences of various features on different scales. Levels of a pyramid are created conventionally in separate steps, i.e., convolution with a separable Gaussian filter followed by decimation.

Suppose the image is represented initially by array g_0, which contains C columns and R rows of pixels. Each pixel is represented as light intensity at a corresponding image point by using integer I between 0 and K. This image becomes the bottom or zero level of the Gaussian pyramid. Pyramid level 1 contains image g_1, which is a reduced or low-pass filtered version of g_0. Each value within level 1 is calculated as a weighted average of values on level 0 within a 5-by-5 window. Each value on level 2, representing g_2, is then obtained from values on level 1 by applying the same pattern of weights, thus obtaining 9 levels.

After filtering, the three features of an image are given values at individual positions based on the input image. These values are divided into 9 levels of pyramid ready to be calculated. Here, the color feature is reflected in two values that we have defined, which are red-green and blue-yellow opponencies. If r, g, b and y are the red,

green, blue and yellow values of the input color image, then the color map for one level is calculated based on the following equations:

$$M_{r-g} = \frac{r-g}{\max(r,g,b)}, \quad \ldots\ldots\ldots\ldots (2)$$

$$M_{b-y} = \frac{b-\min(r,g)}{\max(r,g,b)}, \quad \ldots\ldots\ldots\ldots (3)$$

where M_{r-g}, M_{b-y} represent red-green and blue-yellow opponencies. And $\min(r,g)$ reflects information on yellow, which is perceived as the overlap of red and green in equal parts, meaning that the amount of yellow contained in an RGB pixel is given by $\min(r,g)$. Note that definitions deviate from the original model in [7].

The intensity map on one level is calculated as follows.

$$M_i = \frac{r+g+b}{3}. \quad \ldots\ldots\ldots\ldots (4)$$

These operations are repeated for each level of the input pyramid to obtain an intensity pyramid with 9 levels.

Local orientation map M_o is obtained by applying manipulate filters to intensity pyramid levels M_i [14].

After getting M_{r-g}, M_{b-y}, M_i and M_o, to yield feature maps, we simulate the center-surround receptive fields by subtraction between two maps at center (c) and surround (s) levels in these pyramids. They are calculated as follows.

$$F_{l,c,s} = N(|M_l(c) - M_l(s)|), \quad \ldots\ldots\ldots (5)$$
$$l \in L = L_C \cup L_I \cup L_O,$$

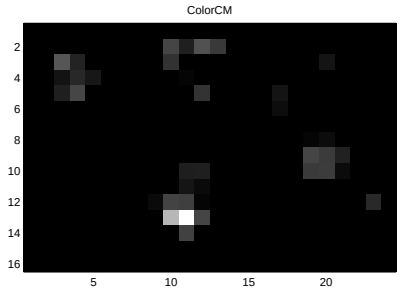
where

$$\left.\begin{aligned} L_C &= \{I\} \\ L_I &= \{r-g, b-y\} \\ L_O &= \left\{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\right\} \end{aligned}\right\}. \quad \ldots\ldots\ldots (6)$$

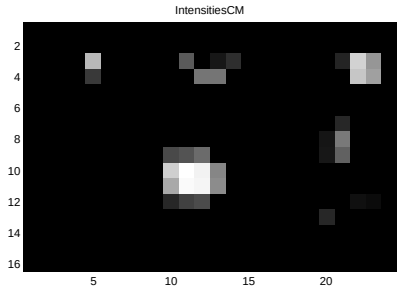
Note that N is an iterative nonlinear normalization operator simulating local competition between neighboring salient locations. Each iteration step consists of self-excitation and neighbor-induced inhibition implemented by convolution with a difference of Gaussians filter followed by rectification. One and five iterations are used for simulations in this paper.

Finally, by summing and renormalizing center-surround combinations based on results from Eq. (6), feature maps of color, intensity, and orientation are obtained from Eq. (7) as C_c, C_i, and C_o. Center-surround receptive fields are simulated by subtraction between two maps at center (c) and surround (s) levels in these pyramids, as shown in Eq. (5). In Eq. (7), for the general features color and orientation, subfeatures contributions are summed and renormalized once more to yield conspicuity maps. For intensity, the conspicuity map is the same as for Eq. (5). **Fig. 11** shows an example of the three feature maps mentioned above.

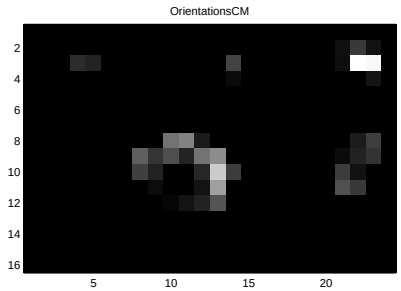
(a) Original image



(b) Color feature map



(c) Intensity feature map



(d) Orientation feature map

Fig. 11. Example of feature maps obtained from image.

$$\left.\begin{aligned} C_i &= F_i \\ C_c &= N\left(\sum_{l \in L_c} F_c\right) \\ C_o &= N\left(\sum_{l \in L_o} F_o\right) \end{aligned}\right\}. \quad \ldots\ldots\ldots (7)$$

4.2. Fuzzy Inference

Fuzzy inference for saliency maps is based on fuzzy logic and resembles human reasoning in its use of approx-

Table 3. Fuzzy rules for saliency map inference.

C	I \ O	OL	OM	OH
CL	IL	SVL	SL	SLL
	IM	SL	SLL	SLL
	IH	SLL	SLL	SM
CM	IL	SL	SLL	SM
	IM	SLL	SM	SM
	IH	SM	SM	SLH
CH	IL	SM	SLH	SH
	IM	SLH	SH	SH
	IH	SH	SH	SVH

imate information and uncertainty to generate decisions. In Section 1, we pointed out a disadvantage of the ordinary combination of feature maps. There is no important distinction between features, especially when one feature is more important than the others.

It has little sense to use fuzzy theory as a classifier in the building stage of feature maps, but using fuzzy theory in the combination stage to infer visual attention regions highlights the significance of various image features and obtains better results. The greatest difference between mathematical summing and fuzzy inference is that the importance reflected in individual feature maps is different. These are tuned by fuzzy rule based on feature values. In fuzzy inference, the importance of a color feature map is higher than for intensity and orientation, to avoid getting unfavorable results if the situation is one in which the color feature is low while the other two are high.

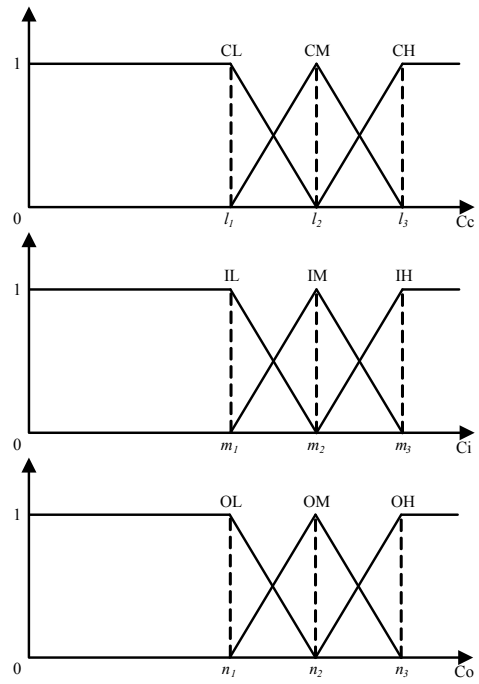
We use feature variables from color feature map (C_c), intensity feature map (C_i), and orientation feature map (C_o) in the *IF* part while the output value in the *THEN* part is the value of region saliency map (S_m). Individual values of region saliency maps are decided by fuzzy rules as shown in **Table 3** and **Fig. 12**. In membership functions in the IF part in **Fig. 12**, values of l, m, and n are used to divide input space into three fuzzy subsets, which are then assigned linguistic terms. We obtained values based on expert experience, setting values based on analysis results of data obtained in the training process. Take l for example, which has the following values.

$$\left.\begin{aligned} l_1 &= 0.4(l_{max} - l_{min}) \\ l_2 &= 0.6(l_{max} - l_{min}) \\ l_3 &= 0.8(l_{max} - l_{min}) \end{aligned}\right\}. \quad \ldots\ldots\ldots \quad (8)$$

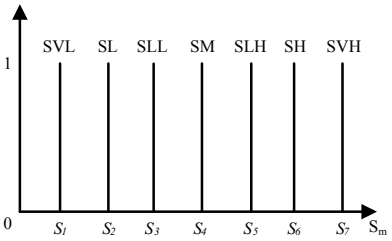
(l_{min}) and (l_{max}) are minimum and maximum values of color feature values.

Fuzzy rules must reflect the importance of the key feature. Fuzzy rules cannot be changed during the inference process, so we obtained the fuzzy rules used in this research based on expert's experience. Fuzzy rules are generated from training input-output pairs.

We proceeded as follows. First, we obtained input and output data by doing numerous experiments using



(a) Membership functions in IF part



(b) Singletons in THEN part

Fig. 12. Membership functions and singletons.

the McGill calibrated color image database as our image database. All images in this database have been calibrated. Using the feature maps calibration method in Section 4.1, we determined the color, intensity, and orientation feature values for individual images. We then divided each input space into three fuzzy subsets and assigned linguistic terms to them. Based on the saliency map of each image, values of individual pixels compose output space. Output space was divided into seven subsets – SVL, SL, SLL, SM, SLH, SH and SVH – representing very low, low, slightly low, intermediate, slightly high, high, and very high. Second, we showed these images to subjects and asked what region they intended to see in the experimental process, that is, they were asked to give three regions in order. The saliency value of the pixel in the users intention region was then raised based on region order. Last, by analyzing data obtained in previous steps, we generated fuzzy rules using assigned linguistic terms. Sample rules are as follows.

- *IF C is low AND I is low AND O is low, THEN S is very low.*

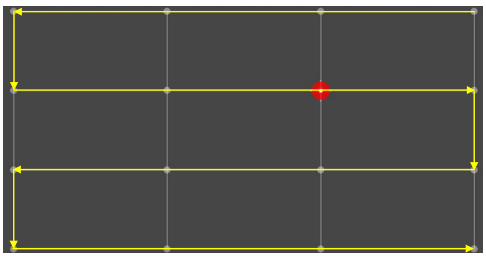


Fig. 13. Position of standard points.

- *IF C is intermediate AND I is low AND O is high, THEN S is intermediate.*
- *IF C is high AND I is intermediate AND O is low, THEN S is slightly high.*

- Fuzzy labels for output on region saliency map (S_m)

  SVL : Very low value of region saliency map
  SL : Low value of region saliency map
  SLL : Slightly low value of region saliency map
  SM : Intermediate value of region saliency map
  SLH: Slightly high value of region saliency map
  SH : High value of region saliency map
  SVH : Very high value of region saliency map

5. Experimental Results

Experiments were divided in two parts. Part 1 consisted of calibrating NN-based eye tracking, which took 24 seconds. Part 2 consisted of asking participants to look at images provided during 1 minute and to name the regions they paid attention to.

During both parts, visual regions and positions in each image, which inferred by using both of the proposed methods mentioned in this paper were recorded based on saliency maps, and gaze positions. We were thus able to replay and analyze each part of experiments.

5.1. Eye Tracking Results

Experiments using our proposed eye tracking were conducted on a notebook computer (Intel Core i3-380M CPU, 2 GB RAM, Microsoft Windows 7 OS). The program was developed in Code Blocks which is an open source IDE and MATLAB R2007a. The pupil center was detected by using OpenCV and openFrameworks, an open source C++ toolkit.

We used 16 points as reference points and asked subjects to stare at each point for 1.5 seconds. In this process, 16 points usually appeared in a 4×4 grid in which so-called "standard" points were prepared and subjects were asked to look at the points when they appeared, as shown in **Fig. 13**. Coordinates of gaze estimation positions were recorded and used as mapping data together with positions of standard points.

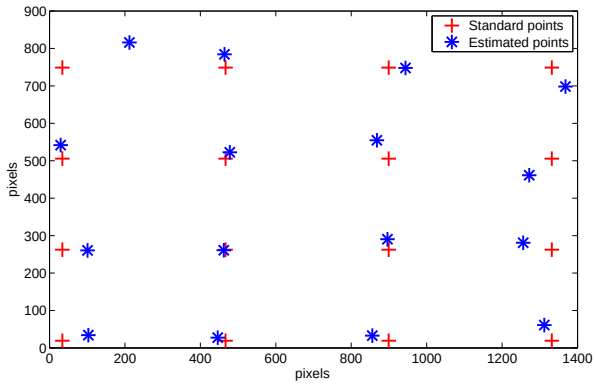


Fig. 14. Experimental results.

Table 4. Eye tracking accuracy.

	X-axis	Y-axis
Distance [%]	3.24	3.68
Direction [deg]	1.172	0.998

The order in which points appeared was based on an "s" type as shown in **Fig. 13**. We found that a subject's behavior easily became set, however, after several experiments and caused a poor calibration result, as mentioned in Section 3.2.2. We therefore use a method by making standard points appear randomly and replacing the previous one. Similarly, when standard points appeared randomly, we recorded both standard and estimation position coordinates. Image acquisition speed was 60 fps, so 1440 (1.5×60×16) points are used as neural network inputs and for output. Because the neural network here is a back propagation (BP) neural network, teaching signals also used the above coordinates. After 1000 trails in the calibration process, actual error matches desired error, 0.001, which is set in **Table 2**.

After calibration, we conducted experiments to verify calibration results. In experiments, 16 points are giving at first as reference points. Next, the subject is asked to look at each point in the sequence. Position data on reference points and the subject's gaze at each position are then recorded simultaneously. Average axes of gaze positions are calculated and plotted in one figure together with reference points. The distance between each estimated point and reference point are shown in **Fig. 14**. The crosses are reference points shown to subjects after calibration process. Asterisks are actual gaze positions on the computer screen from when subjects looked at crosses. The distance between subject's eye and the computer screen in this experiments was 45 cm. Average error in results is shown in **Table 4**. These values indicate that the error between actual gaze position coordinate and intended coordinate is within the acceptable range.

In experiments, the distance between the eye and the computer monitor was 45 cm and screen resolution was 1366×768 pixels, as shown in **Fig. 15**. The width is 28.448 cm and screen height is 21.336 cm. One pixel on

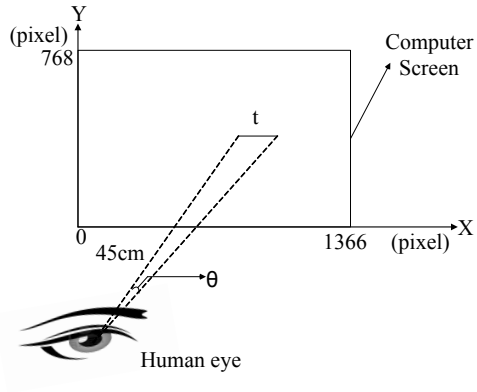


Fig. 15. Degree of error calculation method.

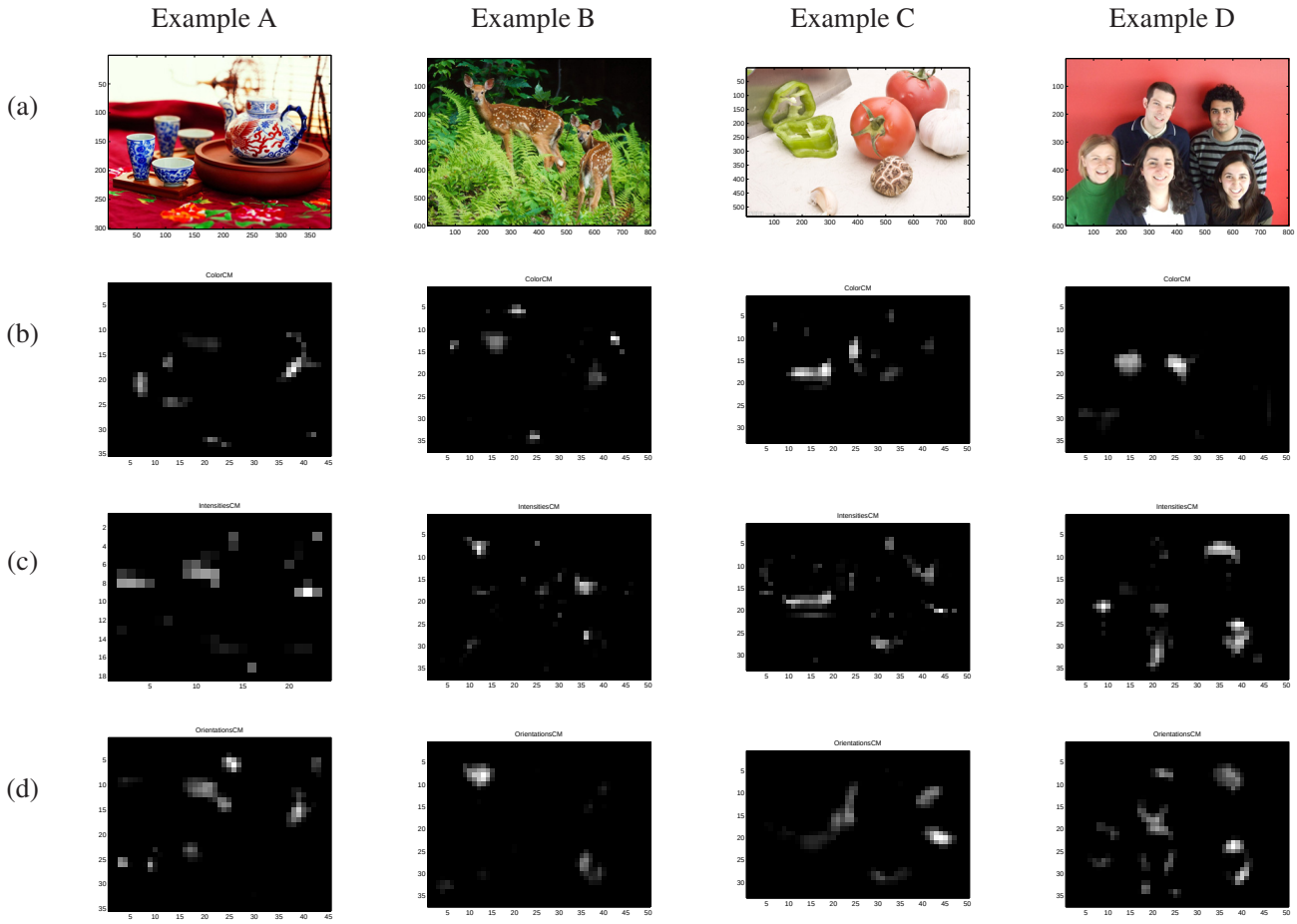


Fig. 16. Four feature maps examples: (a) original image, (b) color feature map, (c) intensity feature map, (d) orientation feature map.

the computer screen thus represents 0.02 cm. The average error in pixels is t, for example, the degree of error is calculated as follows:

$$\theta \approx \arctan\left(\frac{0.02t}{45}\right). \quad \ldots\ldots\ldots\ldots (9)$$

5.2. Saliency Map Inference Results

We conducted several experiments to demonstrate the inference result of the proposed method and to compare the performance of the proposed method with Itti's model [8]. As mentioned in the last section, we first get feature maps. Here, four different images are used as inputs. The input image is processed for low-level features at multiple scales, and center-surround differences are calculated based on Eq. (6). Resulting feature maps are then combined on feature saliency maps based on Eq. (7), as shown in **Fig. 16**.

After feature saliency maps are obtained, the region located on the saliency map is selected for the highest saliency value by proposed fuzzy inference. After seg-

Fig. 17. Four saliency maps examples and attention region: (a) original image, (b) saliency map by sum feature map, (c) saliency map by fuzzy inference, (d) attention region by sum feature map, (e) attention region by fuzzy inference.

mentation around the most salient region location, this saliency map is used to obtain a smooth object mask at image resolution and for showing to subject. Parameters of fuzzy rules (l_i, m_i, n_i) are chosen based on the average value of each feature map. Results for saliency maps and attention regions by summing feature maps and fuzzy inference are as shown in **Fig. 17**. Attention regions are marked by circles and lines show their order.

As seen in the figure, only a few differences appear between saliency maps for the two methods. For example A and B, approximate locations of attention regions and order are basically the same for the two methods, and region shape and size differ little. This is because color, intensity, and orientation all are reflected in regions marked in these two images compared to other regions, which also means that differences of importance for the three features are negligible. The method we proposed did not function very efficiently, but as the results for example C, we can see that the first attention region decided by fuzzy inference is near the tomato, whereas the one decided by summing is near the garlic. This illustrates that our method works because the region nearby the tomato is more conspicuous at the feature at color.

The result for example D shows that attention regions for both methods have obvious differences and that the order is different. From these results alone, we cannot yet recognize whether our proposed method or Itti's is better. In other words, we cannot assert that our proposed method is more accurate for computing a saliency map than the conventional method, so we conducted other experiments to find out.

In these experiments, we asked 5 males between 20 to 30 years old to look at all 50 images while wearing the eye tracking device. As they did so, their gaze positions

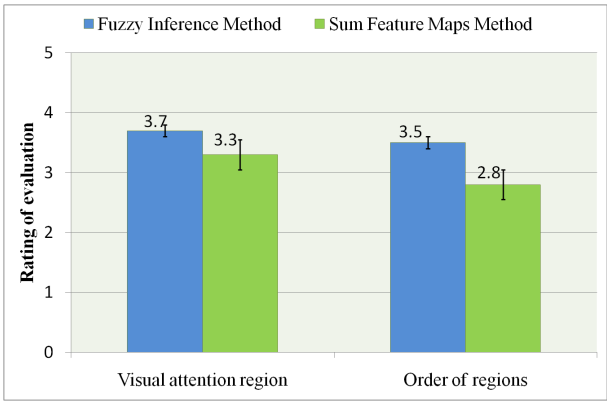


Fig. 18. Standard deviation for evaluation results of attention regions obtained using the two methods.

at different times were calculated and recorded for distribution analysis. After all experiments they will be shown the results for the attention region obtained by the two methods and ask them to compare these with the ones they actually looked at during experiments. Subjects' attitudes toward results were evaluated as shown in **Fig. 18**. Every factor has five ranks represented scores of 1 to 5 from worst to best. Evaluation results indicated that the performance of our proposed method is higher in the order of visual attention region and slightly lower in region detection accuracy. This also illustrates that the proposed fuzzy inference improves the performance of attention region prediction to some extent.

6. Conclusion

We have designed a low-cost, wearable eye tracking system and have proposed new calibration using random calibration points and a neural network to replace conventional spatial mapping in the calibration process. In experiments, we have confirmed that after 1,000 learning trials using enough reference points, better calibration results have been obtained. We have also proposed fuzzy inference method based on saliency maps calculated using color, intensity, and orientation feature maps of images, to predict visual attention regions based on eye tracking. A series of attention region prediction experiments to evaluate the prediction accuracy of our proposed method and results have confirmed the effectiveness of our method in predicting visual attention regions.

One problem still to be soled is that each prediction process for a visual attention region takes about two seconds. This also makes it difficult to apply this method to a real-time system. We now plan to work on improving prediction speed to make it more suitable to real-time prediction.

References:

[1] F. Sadri, "Logic-Based Approaches to Intention Recognition," Handbook of Research on Ambient Intelligence: Trends and Perspectives, 2010.
[2] D. V. Pynadath and M. P. Wellman. "Accounting for Context in Plan Recognition, with Application to Traffic Monitoring," Proc. of the Eleventh Int. Conf. on Uncertainty in Artificial Intelligence, pp. 472-481, 1995.
[3] L. M. Pereira and H. T. Anh, "Intention Recognition via Causal Bayes Networks Plus Plan Generation," Progress in Artificial Intelligence, pp. 138-149, 2009.
[4] K. A. Tahboub, "Intelligent Human-Machine Interaction Based on Dynamic Bayesian Networks Probabilistic Intention Recognition," J. of Intelligent and Robotic Systems, Vol.45, pp. 31-52, 2006.
[5] J. W. Harris and H. Stocker, "Handbook of Mathematics and Computational Science," Springer-Verlag New York, 1998.
[6] W. Mao and J. Gratch, "A Utility-Based Approach to Intention Recognition," Proc. of the AAMAS 2004 Workshop on Agent Tracking: Modeling Other Agents from Observations, 2004.
[7] L. Itti, C. Koch, and E. Niebur, "A Model of Saliency-Based Visual Attention for Rapid Scene Analysis," IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.20, No.11, pp. 1254-1259, 1998.
[8] L. Itti and C. Koch, "A saliency-based search mechanism for overt and covert shifts of visual attention," Vision Research, Vol.40, pp. 1489-1506, 2000.
[9] J. W. Lee, C. W. Cho, K. Y. Shin, E. C. Lee, and K. R. Park, "3D gaze tracking method using Purkinje images on eye optical model and pupil," Optics and Lasers in Engineering, Vol.50, No.5, pp. 736-751, 2012.
[10] A. Bykat, "Convex hull of a finite set of points in two dimensions," Info. Proc. Letters, Vol.7, pp. 296-298, 1978.
[11] I. Mitsugami, N. Ukita, and M. Kidode, "Robot Navigation by Eye Pointing," Proc. Entertainment Computing, pp. 256-267, 2005.
[12] D. Walther, U. Rutishauser, C. Koch, and P. Perona, "On the usefulness of attention for object recognition," Workshop on Attention and Performance in Computational Vision, pp. 96-103, 2004.
[13] L. Itti, "Models of bottom-up and top-down visual attention," Ph.D. thesis, California Institute of Technology, 2000.
[14] W. O. Lee, J. W. Lee, K. R. Park, E. C. Lee, and M. Whang, "Object recognition and selection method by gaze tracking and SURF algorithm," 2011 Int. Conf. on Multimedia and Signal Processing, pp. 261-265, 2011.

Name:
Mao Wang

Affiliation:
Department of System Design Engineering, Graduate School of Engineering, University of Fukui

Address:
3-9-1 Bunkyo, Fukui 910-8507, Japan

Brief Biographical History:
2001-2005 Major of Communication Engineering, College of Physical Science and Technology, Huazhong Normal University
2008-2011 Major of Mechatronic Engineering, College of Mechanical Engineering, Beijing Information Science & Technology University
2011- Department of System Design Engineering, Graduate School of Engineering, University of Fukui

Membership in Academic Societies:
• The Robotics Society of Japan (RSJ)

Name:
Yoichiro Maeda

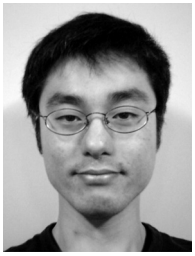
Affiliation:
Department of Robotics, Faculty of Engineering, Osaka Institute of Technology

Address:
5-16-1 Omiya, Asahi-ku, Osaka 535-8585, Japan

Brief Biographical History:
1983- Researcher, Central Research Lab., Mitsubishi Electric Corp.
1989-1992 Senior Researcher, Laboratory for International Fuzzy Engineering Research (LIFE)
1995- Associate Professor, Osaka Electro-Communication University
1999-2000 Visiting Researcher, University of British Columbia (UBC), Canada
2002- Associate Professor, Faculty of Engineering, University of Fukui
2007- Professor, Graduate School of Engineering, University of Fukui
2013- Professor, Faculty of Engineering, Osaka Institute of Technology

Main Works:
• Y. Maeda, M. Tanabe, and T. Takagi, "Behavior-Decision Fuzzy Algorithm for Autonomous Mobile Robots," Information Sciences, Vol.71, No.1, pp. 145-168, 1993.
• Y. Maeda, "Emotional Generation Model for Autonomous Mobile Robot," KANSEI Engineering Int., Vol.1, No.1, pp. 59-66, 1999.
• Y. Maeda and W. Shimizuhira, "Multi-Layered Fuzzy Behavior Control for Autonomous Mobile Robot with Multiple Omnidirectional Vision System: MOVIS," J. of Advanced Computational Intelligence and Intelligent Informatics (JACIII), Vol.11, No.1, pp. 21-27, 2007.

Membership in Academic Societies:
• The Society of Instrument and Control Engineers (SICE)
• The Robotics Society of Japan (RSJ)
• Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT)
• The Japanese Society for Artificial Intelligence (JSAI)
• Japan Society of Kansei Engineering (JSKE)



Name:
Yasutake Takahashi

Affiliation:
Graduate School of Engineering, University of Fukui

Address:
3-9-1 Bunkyo, Fukui 910-8507, Japan

Brief Biographical History:
2000-2009 Assistant Professor, Department of Adaptive Machine Systems, Graduate School of Engineering, Osaka University
2003-2009 Member of exec committee for RoboCup middle size league
2006-2007 Visiting Researcher, Fraunhofer IAIS
2009- Senior Assistant Professor, Department of Human and Artificial Intelligent Systems, Graduate School of Engineering, University of Fukui

Main Works:
• Y. Takahashi, K. Noma, and M. Asada, "Efficient Behavior Learning based on State Value Estimation of Self and Others," Advanced Robotics, Vol.22, No.12, pp. 1379-1395, 2008.
• Y. Takahashi, Y. Tamura, M. Asada, and M. Negrello, "Emulation and behavior understanding through shared values," Robotics and Autonomous Systems, Vol.58, No.7, pp. 855-865, 2010.
• Y. Tamura, Y. Takahashi, and M. Asada, "Observed Body Clustering for Imitation Based on Value System," J. of Advanced Computational Intelligence and Intelligent Informatics, Vol.14, No.7, pp. 802-812, 2010.

Membership in Academic Societies:
• The Robotics Society of Japan (RSJ)
• Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT)
• The Japanese Society for Artificial Intelligence (JSAI)
• The Institute of Electrical and Electronics Engineers (IEEE)