Wearable Vision Interfaces: towards Wearable Information Playing in Daily Life

Norimichi Ukita, Yasuyuki Kono and Masatsugu Kidode

Artificial Intelligence Laboratory, Graduate School of Information Science,
Nara Institute of Science and Technology.

e-mail: {ukita,kono,kidode}@is.aist-nara.ac.jp

Abstract

Our objective is to realize wearable interfaces for expanding our information activities in daily life. To implement these interfaces, we employ vision-based systems, including a variety of camera, a head-mounted display and other visual devices. In this paper, we present two wearable vision interfaces; 1) object registration and retrieval and 2) virtual tablet input by fingertip drawing. In both of these interfaces, an active-infrared image sensor is used to obtain depth-dependent imagery information. This information allows the wearable computing system to easily perform the given tasks (e.g., object analysis and measurement as well as finger motion understanding). Experimental results have demonstrated the effectiveness of the proposed interfaces.

1 Introduction

We have been studying to develop man-machine interfaces for next-generation wearable information playing. To realize these interfaces, we employ vision-based systems, including various cameras, head-mounted display (HMD, in short) and other miniaturized visual devices. A wearable vision system observes the user's viewpoint image with camera(s) and analyses the observed image. Based on the result of the image analysis, the system show the useful information (e.g., images and annotations) to the user. Since an image can 1) give the intuitive information and 2) represent much information simultaneously, we believe that a vision-based interface is appropriate for wearable information playing.

With a wearable vision interface, a user is endowed with the ability for obtaining information acquired with/without a positive interest. We show two examples of the applied systems:

- With a positive interest, a user handles an object in the scene, and then it is observed by wearable camera(s). Its information included in the observed image deserves to be recorded for memory aid, intention analysis, and other man-machine interfaces. Therefore, an object detection, registration and retrieval system is required.
- Without a positive interest also, a user experiences many events in daily life. By always observing and recording the user's viewpoint image, a wearable vision system can provide the image observed at any time. This allows the user to recollect the event which the user has not taken notice of. In addition, the system can extract the image sequence that attracts the user's attention by estimating the gaze history. These properties enable a large variety of functions for automatic accumulation and understanding of the user's information.

We have been aiming at establishing the fundamental technologies for such wearable vision systems.

Especially, we focus on realizing one of the useful wearable vision systems named *Video Diary* (Fig.1), where 1) user's viewpoint images are always observed, 2) the observed image is analyzed to detect the user's attention, 3) and the detected attentions are recored as the personal diary with the observed images and appended information. We consider the video diary to be a typical example for developing various general vision-based interfaces.

In this paper, we present the following two basic vision interfaces:

Object registration and retrieval: The system always detects the object held by the user's hand from the observed image. If the detected object has not been observed, the system registers its information with the observed image. When the user observes the registered object again, the system searches for its image in the stored image sequences.

Wearable virtual tablet: The system detects and tracks an user's fingertip and a plane object in hand, which are utilized as a pen device and an input plane, respectively. While the fingertip traces the plane, the traced locus on the plane is regarded as the input.

We believe that both of the above interfaces are general technologies and can be applied to the video diary and other wearable vision systems.

In sections 2 and 3, we describe the above two interfaces, respectively. Section 4 summarizes the works proposed in this paper and points out future works for next-generation wearable vision interfaces.

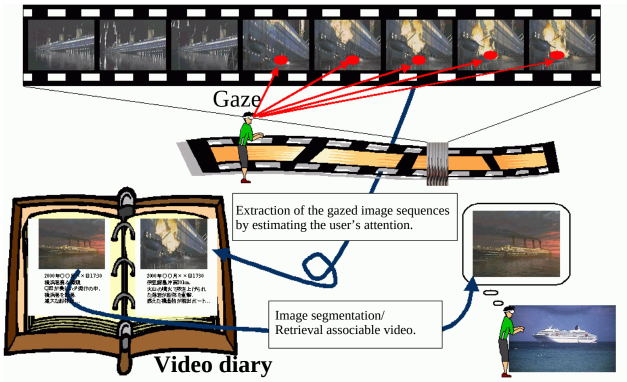


Figure 1: Video diary system.

2 Object Registration and Retrieval System with a Wearable Color and Active-Infrared Imaging Device

In daily life, we use a large variety of objects in the real world. The information about these objects are stored into our memory: object's name, position, and many other attributes. We, however, sometimes forget these information. For this problem, some wearable systems have been developed. Following are two examples of the wearable memory-aid systems:

- In DyPERS[1], a user wears a camera with a HMD on head. If the camera observes a target object, the user stores this object image with its information (e.g., related video and audio information) into a computer. When the camera observes the target object again, the system searches the recorded object images for the current input image and shows the information of the object identified with the current image.
- In [2], we have also presented the wearable memory-aid system similar to DyPERS. The system always stores the continuous image sequences observed from the user's viewpoint while the user walks around in the scene. When the user comes back to the location where the user has been before, i.e., when the current input image is identified with the stored one, the system shows the stored one on the HMD. This assists the user to remember the event that happened in that location. In this system, to cope with the image difference between the input and stored images, the following methods are employed:
  - The image difference caused by head motions are corrected by employing gyro sensors on head.
  - The image difference caused by changes in the scene are ignored by extracting regions of moving objects from the observed image.

In this paper, on the other hand, we focus on how to support the user to recollect the information of the object that is held by the hand. We often hold an object when we are interested in it. For the system to recognize the user's interest in the observed image, object detection and identification are significant. Both of them are, however, difficult to be implemented because there exist many other objects in the image and their 2D positions and postures change depending on the geometric configuration between the user and objects.

We have realized the object detection and identification method that is specialized for dealing with the object in the user's hand. For this method, we 1) developed the wearable camera with color and active-infrared fusion sensors and 2) contrived how to characterize an object image observed by the developed camera for identification. By utilizing this method, we have designed the object registration and retrieval system.

2.1 Wearable Color and Active-Infrared Imaging Device

Here, we present the newly developed wearable imaging device. The device consists of color and active-infrared cameras as illustrated in Fig.2. For wearable information playing, this device has been developed by miniaturizing the similar one introduced in [3]. The prototype system is shown in Fig.3 (a).

The active-infrared camera can be realized with a conventional CCD camera and infrared-pass filter; this camera can capture only infrared rays as a gray scale image as shown in Fig.4 (b). Since infrared rays are irradiated from the illuminant near the device, the camera mainly obtain the infrared rays that are irradiated from the mounted illuminant and reflected from an object in the scene. The obtained reflected rays are captured by the camera as a gray scale image. With the beam splitter, the optical axis of the color camera is calibrated to coincide with that of the infrared camera. In addition, image captured by two cameras are

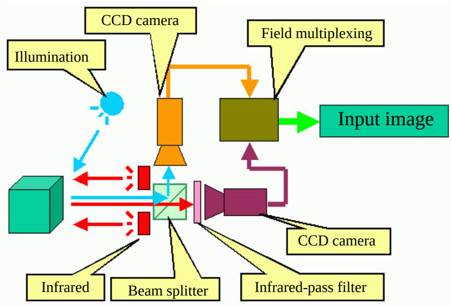


Figure 2: Wearable color and active-infrared imaging device.

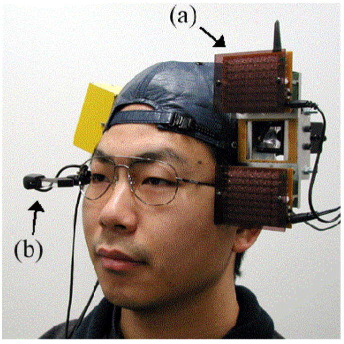
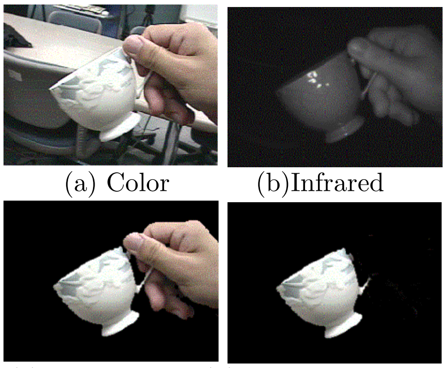


Figure 3: Appearance of the prototype: (a) Imaging device, (b) HMD



(a) Color (b)Infrared

(c) Close object (d) Detected object

Figure 4: Detection of an object region in the observed image.

synchronized because the capturing timing of one camera is activated by the synchronization signal from another camera. Finally, two images are integrated by employing the field multiplexing instrument.

Our prototype system consists of a PC (PentiumIII 600MHz), a wearable infrared camera (ELMO UN411 $\times$ 2 with an infrared-pass filter), light-emitting diodes (LEDs) and a HMD (MicroOptical CO-1). The sizes of both color and infrared images are 320 $\times$ 240[pixel].

Figures 4 (a) and (b) show the observed color and infrared images, respectively. The intensity of the reflected infrared rays depends on the distance from the illuminant to objects in the scene. By employing this property,[1] close object regions can be easily detected from the color image (Fig.4 (c)) without complicated methods for 3D depth reconstruction.

Next, regions corresponding to th user's hand has to be removed from the close object regions. We implement this procedure by eliminating the skin color regions. We determine the skin color in HSV-color color space[2] in advance. Figure 4 (d) shows the result. As we can see, the system can work well.

2.2 Object Registration and Retrieval System

With the developed imaging device, we design the object registration and retrieval system.

2.2.1 Basic Scheme for Object Registration and Retrieval

Following are the tasks of the system:

1. Observe an object in the user's hand.
2. Detect the object region from the observed color and infrared images.
3. Extract the object information required for identification.
4. Register the extracted object information with its image if the object has not been observed.
5. Otherwise, search for the registered object image corresponding to the object image in the current input image. If this object identification is successful, the identified object image is shown on a HMD illustrated in Fig.3 (b).

In what follows, we present each function for implementing the above tasks.

2.2.2 User Operation while Registration

When the user holds an object in hand, it is observed by the camera. Then, the object regions in the observed image are detected based on the object extraction described in Sec.2.1. Depending on object's position and posture, the appearance of the detected object varies. This results in difficulty in object identification. Basically, to cope with this problem, object images observed from various viewpoints are required. In our system, therefore, the user moves an object while observing it if its information has not been registered, namely if object identification between it and registered objects are failed (mentioned later).

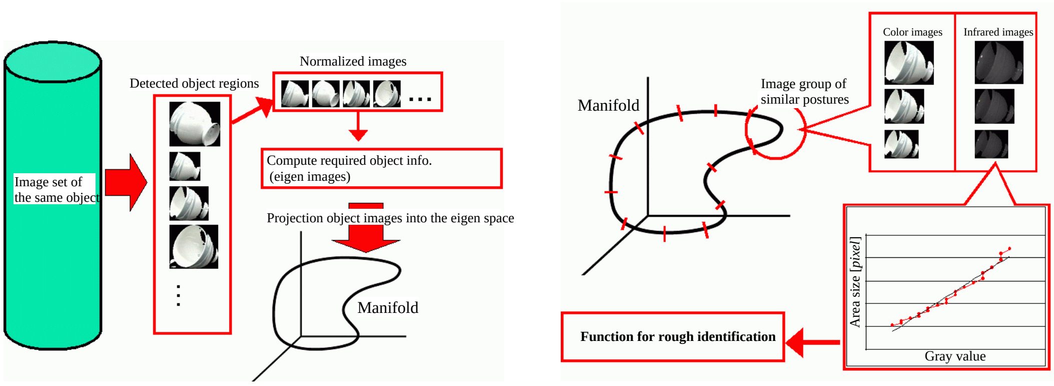
2.2.3 Discrimination between Similar Objects using Infrared Image

Basically, object identification methods can be categorized into two classes:

Appearance-based methods: Appearance features of an object is analyzed. In [4], all object images observed from various positions and postures are recorded as a manifold in the eigen space. In [5], an object image is represented as a set of local features that are invariant to illumination changes and affine or 3D projection.

[1] To determine close object regions, we give a threshold to discriminate between close and background regions in advance.

[2] Since a hue value in the HSV-color space changes little even if illuminations varies in the scene, we can extract the skin color region robustly against illumination variations.



(a) Manifold in the eigen space (b) Grouping analogous posture images

Figure 5: Acquisition of object information required for identification.

3D reconstruction methods: 3D volumetric information of the object is reconstructed and compared with that of another object. In [6], a rough 3D volume of a free-form object is reconstructed using segment-based stereo vision.

In the former method, since all object images are normalized, it is impossible to discriminate between similar objects, namely objects whose appearances are identical but dimensions are different. This problem is avoidable by employing the latter method with the former appearance-based method. The 3D reconstruction, however, 1) is sensitive to image noises and 2) requires complicated processings and computations. These properties damage robust identification.

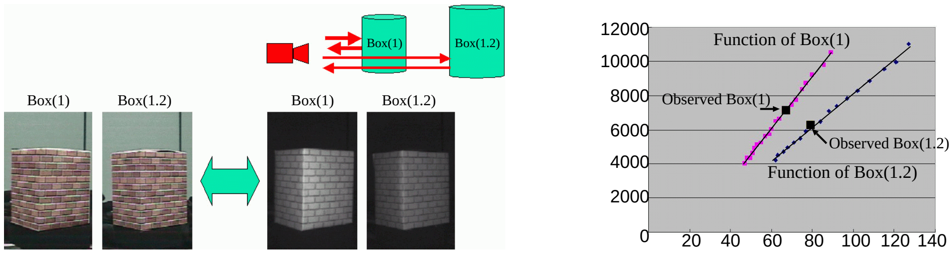
We present a simple but effective object identification using color and infrared images. As mentioned before, the intensity of gray values in an infrared image depends on the distance between the camera and the observed object. With this property, we can easily discriminate similar objects.

However, the intensity of infrared illumination also depends on the surface shape and the reflection coefficient of the object. This means that gray values corresponding to a single object in the infrared image are different from each other depending on the position and posture of the object. We, therefore, classify the images of a single object into sets of images whose objects' postures are analogous. For this classification, we utilize the parametric eigen-space method proposed in [4]. This method projects the images of an object to its eigen space (Fig.5 (a)). In the eigen space, the images of analogous appearances neighbor on each other. Based on this neighboring relations, a set of images corresponding the analogous postures of the object can be determined (Fig.5 (b)). We call this set and the images in the same set the *Analogous Posture Set* and the *Analogous Posture Images*, respectively.

For each similar posture images, the relation between area sizes of and gray values in the infrared images is estimated (Fig.5 (b)). We call this relation the *Rough Identification Function*. Based on this function, similar objects can be discriminated as follows. When an object region in the observed image is detected, its area size and gray value in the infrared image (denoted by A_o and G_o, respectively) are compared with the rough identification function of the recorded object. If (A_o, G_o) is not matched with the function, namely the distance between (A_o, G_o) and the functional curve is larger than the predefined threshold, identification between two objects is unsuccessful. This method is applicable not only to discriminate similar objects but also to narrow down the candidates for object identification.

Here, we show the experimental results that demonstrate the effectiveness of our proposed identification method. In this experiment, we observed two boxes whose textures and shapes were similar to but sizes were different from each other. Figure 6 (a) shows the observed color and infrared images. The size of the right box (called *Box(1.2)*) was 1.2 times as large as that of the left box (called *Box(1)*). However, since the distances between these boxes and the camera were adjusted, the area size of Box(1) was equal to that of Box(1.2) in the observed image as shown in Fig.6 (a).

Figure 6 (b) indicates the relations between area sizes of the images and gray values in the areas. First, the system extracted the object region from the infrared image, then computed its area size[*pixel*]. To avoid the influence of noises and specularities in the image, the median gray value in the object region was regarded as the typical gray value. In this experiment, we determined the rough identification functions of these boxes as the linear equations (denoted by black lines in Fig.6 (b)). The newly observed information of Box(1) and Box(1.2) were compared with two functions; the distance between the projected point corresponding to the newly observed object and the line of each function is evaluated (Fig.6 (b)). If the distance between the projected point and the line of a registered object (denoted by object_r) is larger than the predefined threshold,



(a) Observed objects and its images (b) Rough identification function

Figure 6: Experimental results.

(a) Object 1 (b) Object 2 (c) Object 3

Figure 7: Examples of the observed images of three objects.

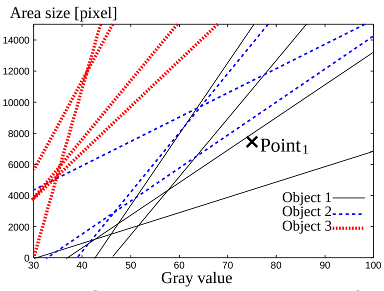


Figure 8: Obtained rough identification functions.

the newly observed object is not identified with object$_r$. That is, object$_r$ is excluded from candidates for the next identification procedure. From these results, it can be confirmed that the proposed method is simple but effective enough to discriminate between similar objects that are almost indistinguishable.

2.2.4 Object Identification for Image Retrieval

Successful rough object identification does not mean the final result; this identification does not take into account the shape, color, and other appearance attributes of the observed object. To obtain the final result, identification using the eigen-space method follows on rough object identification. Object identification using the subspace method[7] is implemented as follows:

1. Suppose rough object identification between the detected object region in the observed image and the registered objects (denoted by object$_{1,\cdots,N}$) are successful.
2. Project the image of the detected object to the subspace of object$_{1,\cdots,N}$ to evaluate the correlation.
3. Let 1) object$_i$ have the highest correlation with the detected object and 2) the correlation be higher than the predefined threshold. The system considers the detected object to be object$_i$.

If successful identification result is not obtained, the current observed object is considered to be newly detected object. Then, the system recommends the observation operation described in Sec.2.2.2 to the user for registration.

We conducted experiments to verify the effectiveness of the proposed system. We registered three objects shown in Fig.7. Then, the rough identification function and subspace for each object were generated. When the system observed object 1, the object information extracted from the observed image (denoted by Point$_1$ in Fig.7) was first compared with rough identification functions as illustrated in Fig.8. Since no lines of object2 has a short distance from point$_o$, it is eliminated from the candidates for identification using the subspace method. Then, the subspace method identified the image of the detected object with the registered information of object$_0$. In intensive experiments, each object could be successfully identified.

As we can see, the experimental results demonstrate that the proposed system work well to 1) detect and register the information of the object in the user's hand and 2) identify the observed object with the registered object for retrieval.

3 Wearable Virtual Tablet: Fingertip Drawing Interface using Infrared Image

3.1 Free-content Input Interface for Wearable Computing

To input information into a computer, we need an interface device. There exist many interface devices for desktop computers (e.g., keyboard, mouse, tablet, and so on). For wearable computing, however, a portable interface that can be used at anytime, anywhere is required.

As wearable technology advances, several works about portable interfaces have been reported:

FingeRing[8]: By typing with fingertip action that are detected by ring shaped sensors on each finger, an user can virtually use a keyboard.

WristCam[9]: With a camera attached under the user's wrist, finger motions are observed. This allows us to unobtrusively input gestures to a computer.

HIT-Wear[10]: With a camera mounted near the user's viewpoint, the shape/gesture of the hand/fingers are recognized. Then, a selective menu is superimposed on the hand region in the observed image displayed on the HMD.

While these interfaces are useful for specifying predefined characters and choices, it is impossible to input free contents, namely arbitrary information (e.g., pictures and symbols).

Although the following free-content input systems have been also developed, they have some disadvantages for a wearable convenient interface:

Pen device with physical sensors[11]: A user possesses a pen device with gyro and acceleration sensors. By employing these sensors, the 2D locus of the pen device in the air can be tracked. This system, however, has the following disadvantages:

- The user has to always possess the special pen device.
- It is difficult for us to intuitively understand how to translate our motion into the 2D locus by the sensors. That is, the written locus is often different from the user's design because of the complex mechanical characteristics of the sensors and writing without any input plane.[3]
- The user has to turn on/off a mechanical button for switching the input state from on/off to off/on.

Fingertip writing with a single stroke[12]: A user writes a character with a single stroke by moving the fingertip in the air. The finger motion is observed by the head-mounted camera, and the 2D fingertip locus is displayed on the HMD. Although this system is used only for character input, the basic technology for drawing with the fingertip is applicable to the free-content input interface. This system, however, has an essential problem; the user writes a character with a single stroke, namely the fingertip motion is always regarded as the input locus after the user's finger has been once detected. To stop the input, the user has to get the finger out of the visual field of the camera. This troublesome operation incurs a degeneracy of input.

To solve the problems in these systems, the system should provide the following functions simultaneously:

- Draw the locus by the fingertip.
- Switch the input state from on/off to off/on without any mechanical switches.

To realize these functions, we propose the *Wearable Virtual Tablet* (WVT, in short), where a user can draw a locus on a plane object with the fingertip.

3.2 Wearable Virtual Tablet and Its Architecture

The WVT system consists of a wearable camera with a HMD. In the WVT system, we draw an arbitrary locus on an arbitrary plane object even while freely walking and moving our body. The position of the fingertip on the input plane is detected and tracked in the observed image. Then, its locus is superimposed on the current observed image, and the superimposed image is shown in the HMD to support the user to continuously draw. We believe that the WVT is a basic technology for free-content input.

Following are the technical problems to realize the WVT:

Problem 1 Determine the input plane.

Problem 2 Detect and track the user's fingertip.

Problem 3 Discriminate between the input and non-input states.

Problem 4 Superimpose the locus of the fingertip on the current observed image.

In this paper, to make problem 1 easy, an arbitrary rectangular plane held by the user's hand is regarded as an input plane. The system has to detect its four corners to determine the area in the image observed by the wearable camera. We employ an infrared camera introduced in Sec.2 as the wearable camera. Its properties drastically simplify the detection of the input plane as well as problem 2; the infrared image enables the system to easily detect the input plane and the user's finger without being confused by complicated background scene and textures on the input plane. Furthermore, the gray-scale information in the infrared image allows the system to estimate whether or not the fingertip touches the input plane. Finally, problem 4 can be realized by translating the fingertip locus between the current and buffer images; based on the geometric configuration among four corners of the input plane, the projection between two tetragons can be computed.

In what follows, we describe how to realize the WVT practically.

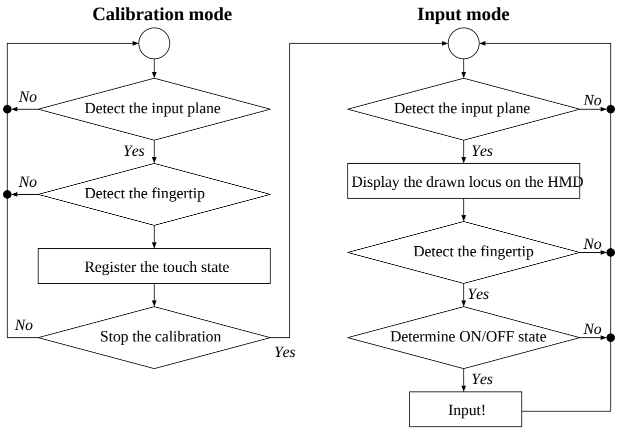
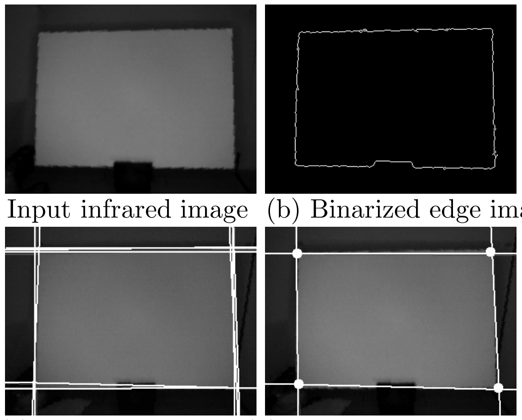


Figure 9: Flowchart the WVT.



(a) Input infrared image (b) Binarized edge image

(c) Detected lines (d) Four sides and vertices

Figure 10: Detecting the vertices of the input plane.

3.3 Fingertip Drawing on Input Plane

Figure 9 shows the flowchart of the WVT. The WVT has two functional modes, namely the calibration and input modes:

Calibration mode: To determine whether or not a user is drawing, a distribution of gray values around the fingertip in the infrared image is registered in advance. For this calibration, the above problems 1 and 2 have to be solved.

Input mode: After the calibration mode, a user starts drawing. In this mode, the above problems 1, 2, 3 and 4 have to be solved.

3.3.1 Detection and Tracking of Input Plane

In our system, four vertices of an input plane have to be detected to determine the area of the input plane and superimpose the drawn locus in the observed image. Suppose objects close to the user are only the input plane and the user's hand. As mentioned before, the system can easily extract regions corresponding to close objects in the infrared image (shown in Fig.10 (a)) without being interfered by complicated background scene and textures on the input plane.

Edge detection, binarization and erosion are applied to the close-object image. The result is shown in Fig.10 (b). To detect four sides of the input plane, we employ the Hough transform[14]. The detected result is shown in Fig.10 (c). If similar lines are detected shown in Fig.10 (c), these lines are bundled up and become a single line. Then, the four lines are determined. Lastly, the intersection points of these lines are considered to be the four vertices (Fig.10 (d)).

3.3.2 Detection and Tracking of Fingertip

The system searches for a fingertip position in the binarized edge image. To simplify the task, we suppose that people point the fingertip upward in the observed image while drawing. Then, the fingertip can be detected by the following processes:

1. If the close-object region is detected in the input plane area, the system regards it as the region of the user's hand.

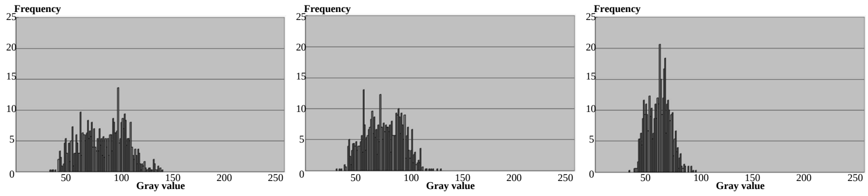
2. By raster-scanning in the input plane area, the edge region is searched in the binarized edge image.

3. The firstly detected edge region is regarded as the fingertip.

3.3.3 Discrimination between Input and Non-input States

To register the variation of gray values around the fingertip in the case of touch, the system observes the input plane while the user is drawing on it. During this operation, the system detects the fingertip region and acquires the histogram of gray values in the edge image. Figure 11 (a) shows an example of the acquired histogram. For comparison, we show also examples of the gray-value histograms in the case of non-touch (Fig.11 (b) and (c)), both of which were observed in the same condition, i.e., 1) the same position and posture of the input plane and 2) the same position of the fingertip in the observed image. The distributions of gray

[3] [13] reports that a touch feeling allows us to efficiently write and type.



(a) 0[cm] (touch) (b) 3[cm] (non-touch) (c) 5[cm] (non-touch)

Figure 11: Histograms of the gray values around the fingertip.

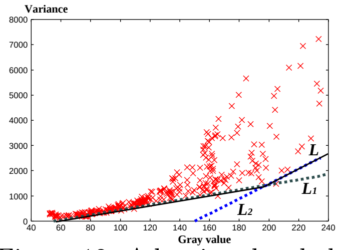


Figure 12: Adaptive threshold.

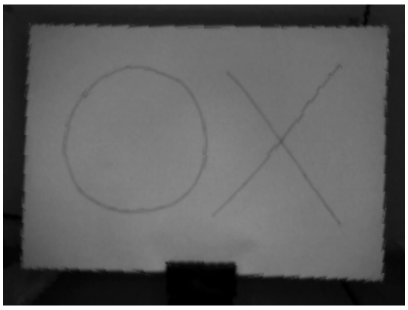
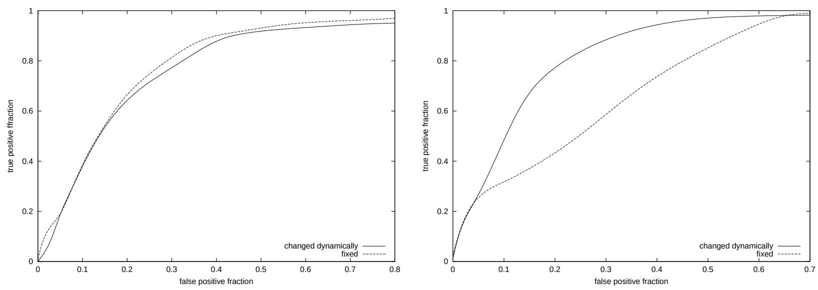


Figure 13: Input plane object.



(a) Perpendicular plane (b) Slanted plane

Figure 14: ROC curves for two experiments.

values are different from each other. We characterize the distribution as a variance of gray values. In these experimental results, the closer the distance between the fingertip and the input plane becomes, the larger the variance of gray values becomes. In our system, therefore, the input state is detected if the variance of gray values is larger than the predefined threshold. The smallest variance acquired during the above operation is considered to be the threshold.

Note that gray values in the input plane vary depending on its position and posture. This means that the fixed threshold fails the detection of the input state if the input plane moves. To solve this problem, we adjust the threshold depending on the average of gray values around the fingertip. The relation between the threshold and gray values around the fingertip is shown in Fig.12. The horizontal and vertical axes of the graph show the average of gray values and the variance in the edge image, respectively. This graph was obtained by observing the fingertip on the input plane while the position of the fingertip and the geometric configuration between the camera and the input plane were changed. In the graph shown in Fig.12, we approximately represented the lower boundary of the points as two linear equations represented by the bended line L that consists of lines L_1 and L_2. If the projected point of the observed fingertip is above L when the system is in the input mode, the system considers that the fingertip touches the input plane.

3.3.4 Continuous Superimposed Display of Input Locus

To correctly transform the fingertip locus drawn in the input plane between different observed images, we have to estimate the 3D geometric configuration between the camera and the input plane. This procedure, however, requires complicated reconstruction of 3D information.

In the proposed system, therefore, the following transformation is employed:

$$P(u, v) = (1 - v)((1 - u)P_{00} + uP_{10}) + v((1 - u)P_{01} + uP_{11}), \tag{1}$$

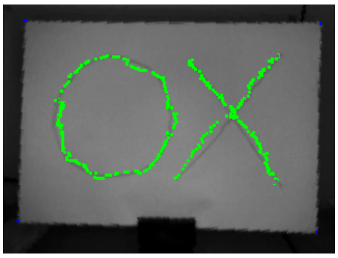
where $0 \leq u \leq 1$, $0 \leq v \leq 1$ and P_{00}, P_{01}, P_{10} and P_{11} denote a 2D point. This equation transforms a point in the square (denoted by B_s) determined by $(0, 0)$, $(0, 1)$, $(1, 0)$ and $(1, 1)$ to the point in an arbitrary tetragon determined by P_{00}, P_{01}, P_{10} and P_{11}.

We regard B_s as the image buffer for recording the fingertip locus. The continuous superimposed display of the fingertip locus on the HMD is implemented by the following procedures:

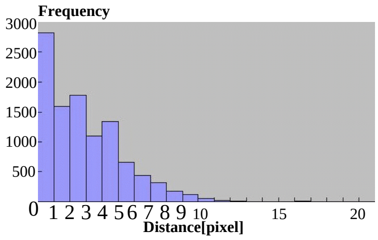
- When the input result (i.e., the fingertip position on the input plane) is newly obtained, it is projected to the image buffer by the inverse projection of Equation(1).
- At every capturing timing, the locus in the image buffer is reprojected onto the observed image by employing Equation(1).

3.4 Experiments

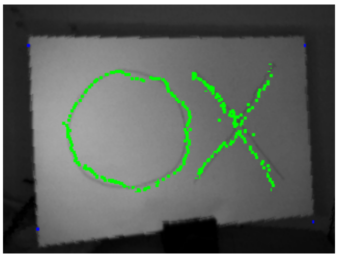
We conducted experiments to verify that the proposed system work well. Our system consists of a PC (PentiumIII 1GHz), an active-infrared camera and a HMD. The size of a captured image is 320 × 240. In all



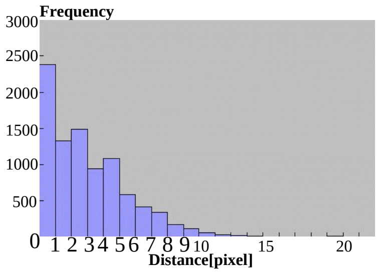
(a) Displayed superimposition



(b) Histogram



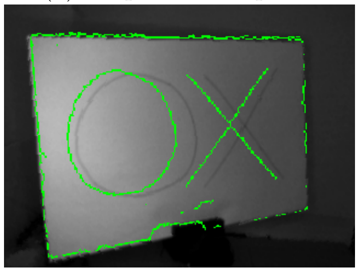
(c) Displayed superimposition



(b) Histogram

Figure 15: Input accuracy. Upper: Input plane perpendicular to the optical axis of the camera, Lower: Slanted input plane.

(a) Perpendicular plane



(b) Slanted input plane

Figure 16: Superimposition while moving the plane.

experiments, we used the A4-size paper shown in Fig.13 for an input plane. To verify the performance of the proposed system, the reference figures (i.e., the circle and the cross) are drawn on the input plane object.

In what follows, we show three experimental results, which shows 1) the correctness of the discrimination between input and non-input states, 2) the input accuracy, and 3) the effectiveness of the superimposed display.

The user tracked the figure on the input plane with the fingertip. We assume that the ground truth of the drawn fingertip-locus is identical to the reference figure. If the distance between the position of the detected fingertip-input and the reference figure is smaller than 5[pixels], this input is regarded as the correct input. Otherwise, this input is regarded as the error input. The former and latter mean "the system correctly detects the input when the user touches the input plane" and "the system detects the input by mistake when the user does not touch the input plane", respectively. To verify whether or not correctly discriminate between the input and non-input states, we evaluated the following two rates in the ROC curve:

True positive: The rate of correct inputs.

False positive: The rate of error inputs.

We evaluated these values for the proposed adaptive threshold method (solid line in Fig.14) and the fixed threshold (dotted line in Fig.14). To obtain the ROC curves for the proposed and fixed-threshold methods, 1) line L in Fig.12 was translated along the vertical axis and 2) the threshold was gradually adjusted, respectively. Figures 14 (a) and (b) show the results for the plane perpendicular to the optical axis of the camera and the slanted plane, respectively. For input in the slanted plane, the advance of the proposed method was remarkable. Accordingly, it can be confirmed that the proposed adaptive threshold method is required for wearable computing environment.

Next, we verified the input accuracy by evaluating the error distance between the reference figure and the position of each input. The user tracked the reference figure with the fingertip as well as in the above experiment. Figures 15 (a) and (c) show the drawn loci. Figures 15 (b) and (d) show the histograms whose horizontal and vertical axes indicate the error distance and the number of the drawn points. In the perpendicular plane, the average, median and variance values of the error distance were 3.06, 2.83 and 5.64, respectively. In the slanted plane, 3.15, 2.83 and 6.15. In both cases, the rates of the points, whose error distances were within 5[pixel], were over 90%. We consider this accuracy to be enough for practical use.

Lastly, to verify the effectiveness of the superimposed display, we projected the reference figure from the perpendicular plane (shown in Fig.16 (a)) to the slanted plane(shown in Fig.16 (b)). The projected figure did not completely coincide with the reference figure on the slanted plane. We can, however, continuously draw in the slanted input plane while visually checking the geometric configuration between the projected figure and the newly input locus.

4 Concluding Remarks

This paper presented two wearable vision interfaces, namely the object registration/retrieval and virtual tablet.

Wearable object registration/retrieval system With newly developed wearable color and active-infrared imaging device, the object registration and retrieval system was realized. This system records the object image extracted from the observed image into the wearable PC. The recorded object image is shown on the HMD when the user holds and observes the object again.

Wearable virtual tablet By employing the properties of the active-infrared camera, we developed the wearable virtual tablet. The user can draw an arbitrary locus on a rectangular plane held by the user's hand. Since the drawn locus is displayed on the HMD and its shape is dynamically adjusted depending on motions of the user and input plane, the user can continuously utilize the WVT in wearable computing environment.

By integrating the developed two interfaces, more useful system that registers and retrieves an object image with its information can be realized; 1) we can input free-content information about the observed object by the WVT and register the input information with the observed image, and 2) retrieve the registered image in accordance not only with a newly observed image but also with object information inputed by the WVT.

Acknowledgements

We would like to thank T. Ueoka and A. Terabe in our laboratory for useful discussions and intensive experiments.

References

[1] T. Jebara, B. Schiele, N. Oliver and A. Pentland, "DyPERS: Dynamic Personal Enhanced Reality System", Perceptual Computing Technical Report #463, MIT Media Laboratory, 1998.

[2] T. Kawamura, Y. Kono and M. Kidode, "A Novel Video Retrieval Method to Support a User's Recollection of Past Events for Wearable Information Playing", in *Proc. of The Second IEEE Pacific-Rim Conference on Multimedia*, pp.24-31, 2001.

[3] H. Sasaki, T. Kuroda, Y. Manabe and K. Chihara, "Hand-Area Extraction by Sensor Fusion Using Two Cameras for Input Interface of Wearable Computers", in *Proc of The 12th Scandinavian Conference on Image Analysis*, pp.779–784, 2001.

[4] H. Murase and S. K. Nayar, "Visual learning and recognition of 3D objects from appearance", International Journal of Computer Vision (IJCV), Vol. 14, No. 1, pp.5-24, 1995.

[5] D. G. Lowe, "An object recognition system using local image feature of intermediate complexity", in *Proc. of International Conference on Computer Vision (ICCV 99)*, pp.1150–1157, 1999.

[6] Y. Sumi, Y. Kawai, T. Yoshimi and F. Tomita, "Recognition of 3D Free-Form Objects Using Segment-Based Stereo Vision", in *Proc. International Conference on Computer Vision (ICCV 98)*, pp.668–674, 1998.

[7] E. Oja, "Subspace Methods of Pattern Recognition", Research Studies Press, Hertfordshire, 1933.

[8] "Body Coupled FingeRing: Wireless Wearable Keyboard", ACM CHI'97 Proceedings, pp.147-154, 1997.

[9] L. T. Cheng, J. Robinson and A. Vardy, "The Wristcam as Input Device", in *Proc. of International Symposium on Wearable Computing (ISWC 99)*, pp.199–202, 1999.

[10] H. Sasaki, T. Kuroda, Y. Manabe and K. Chihara, "HIT-Wear: A Menu System Superimposing on a Human Hand for Wearable Computers", in *Proc. of International Conference on Artificial Reality and Teleexistence (ICAT 99)*, pp.146–153, 1999.

[11] Y. Yamamoto and I. Shiio, "A Simple AR System for Casual Communication", in *Proc. of Workshop on Interactive Systems and Software (WISS 2000)*, pp.117–124, 2000. (Written in Japanese)

[12] Y. Muraoka and T. Sonoda, "A Letter Input System of HandWriting Gesture - A User Interface for Wearable Computers -", in *Proc of Interaction 2001*, pp.3–10, 2001. (Written in Japanese)

[13] J. K. Hahn, J. L. Sibert and R. W. Lindeman, "Towards Usable VR: An Empirical Study of User Interfaces for Immersive Virtual Environments", in *Conference on Human Factors in Computing Systems (CHI 99)*, pp.64–71, 1999.

[14] H.Koshimizu and M.Numada, "On a Basic Consideration of the Warp Model of Hough Transform", in *Proc of Machine Vision Application (MVA'92)*, pp.7–9, 1992.