Technology Reports

Special Articles on User Interface Research —New Interface Design of Mobile Phones—

"Nandemo Interface" — HMD Allowing Content to be Viewed and Operated Using Nearby Objects

As part of realizing a "Nandemo Interface," we have developed a system which uses an HMD and allows nearby objects, such as note books or file folders, to be used like a touch-panel display, to view content and perform input operations. With this system, users can view content displayed on a nearby object of their choosing and perform touch-panel-like input operations on it. This makes it possible to enjoy online shopping, TV and Internet content anywhere, at any time, and at any desired screen size.

Research Laboratories

Mikiko Nakanishi
Tsutomu Horikoshi

1. Introduction

One of the major difficulties with mobile terminals is the small size of their screens. According to market surveys, even though mobile terminals sold recently have much larger screens than earlier models, they are too small, and video viewed on the screen is difficult to see [1]. They are still not large enough for viewing content without stress. Nevertheless, considering that mobile terminals are normally carried around, they cannot be given a screen as large as a PC or TV.

In this article, we propose a "Nandemo Interface" (nandemo means "anything" in Japanese), which allows users to view content at any size without having to carry around a large display, by using nearby objects such as a wall or a note book that may be in their briefcase ("anything"), as the display. We also describe a prototype system that uses a Head Mounted Display (HMD)[*1] to implement this concept.

2. Nandemo Interface Overview

The prototype system uses a display, called an HMD, which is worn like a pair of glasses (**Figure 1**), and this particular HMD is a video see-through type[*2] display.

There are various examples of research on systems that allow viewing of content and input/output operations without using any objects, including systems that display an image of the content (a "content image") floating in the air [2], and systems that perform input/output operations using gestures in the air [3].

However, when we perform input operations and view content in everyday life, we always use some object as an input/output device. For example, we use a keyboard for entering text, and

---

HMD
Recognize locations of file folder and hand using the camera

Image seen through the HMD

back

Marker
Attached to the file folder so it is recognized for use as a display

Finger-ring type microphone
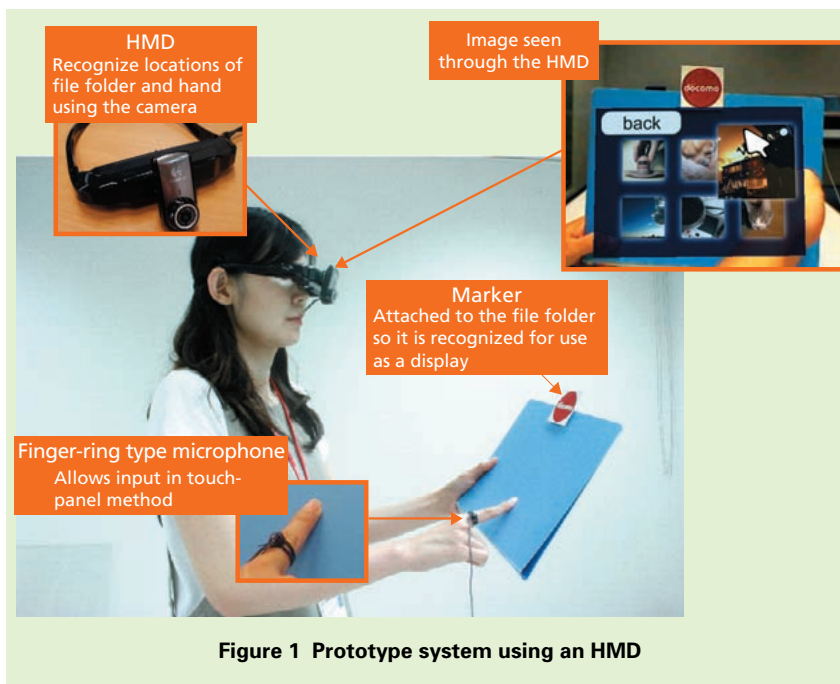Allows input in touch-panel method

Figure 1 Prototype system using an HMD

a display for viewing content. Since we normally perform input operations and view content in this way, new methods that do not use objects present issues such as no button-press sensory feedback when inputting, or ambiguity in where the display can be touched. Some experimental results indicating that using real objects is effective for viewing content and performing input operations using an HMD have also been reported [4]. For these reasons, we decided to use objects at hand for displaying content and for input operations for our system.

The HMD used for the Nandemo Interface we have developed has an integrated camera that we use to recognize the object (such as a file folder or notebook) to be used as the display. The content image is overlaid on it,

adjusting to the size of the object (a file folder) as shown in Fig. 1. In this way, the object held by the user appears as a display, and the size of the content image can be adjusted freely by adjusting the size of the item used.

Also, user input operations can be implemented by using the camera to recognize the position of the user's hand and a finger-ring type microphone to detect the sound produced when the user taps the content image.

3. Nandemo Interface Implementation

3.1 Overall View of the Prototype System

An overall view of the system is shown in **Figure 2** (a). PC1 processes the audio from the finger-ring type microphone and the image from the

camera, performs object recognition and detects user input operations, and executes drawing processing to overlay the content image over the camera image. The output image from PC1 is displayed as-is on the HMD.

PC2 is used as the content server, storing and managing content which the user can view through PC1.

When the user performs an input operation, the finger position and time are sent from PC1 to PC2 and the content image display is rewritten.

Currently, PC1 is used as the terminal (or mobile terminal) and PC2 is the content server, but in future systems, the camera and microphone of a mobile terminal can be used as shown in Fig. 2 (b), and other processing can be done over the network on servers, in a thin client[*3] architecture.

3.2 Overview of Processing in the Prototype System

The user attaches a marker to the object to be used as the display, enabling it to be used for viewing content and for input operations. The specific processing performed to implement this is described below.

1) Detecting Marker Position and Orientation

In order to use a nearby object as a display, the system must be able to recognize the object that the user wants to use. For our system, we have used a marker to find this specific object within the camera image (**Figure 3**). It is

(a) Current system

Finger position and time touched object

Content image

PC2

PC1

Camera image, finger-ring type microphone sound

Image with content overlayed on camera content images (image shown on HMD)

・Capture of finger-ring type microphone sound and camera image
・Object recognition
・Detection of user input operations
・Drawing

(b) Future-system concept

Content-provider content server

Internet

Home PC or appliance

・Object recognition
・Detection of user input operations
・Drawing

Capture of finger-ring type microphone sound and camera image

NTT DOCOMO facilities

Wireless

Camera image, finger-ring type microphone sound

Image shown on HMD

Mobile terminal

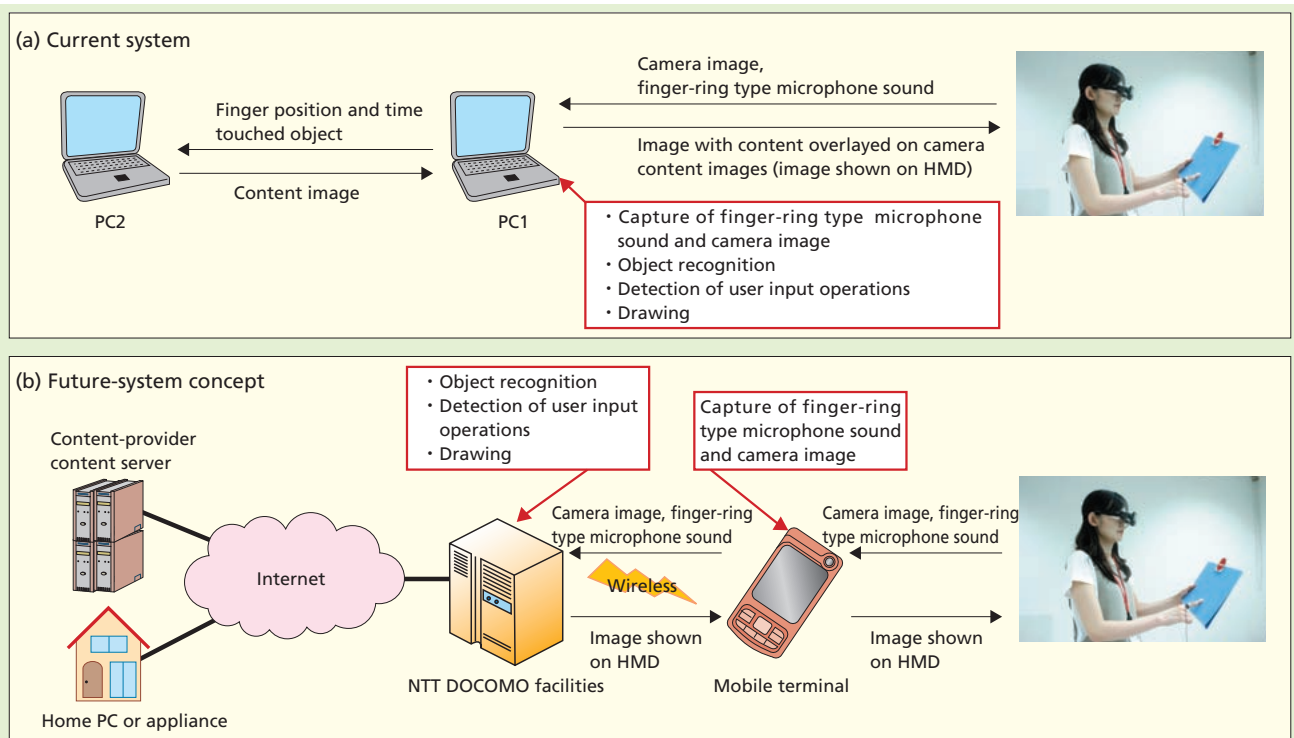Camera image, finger-ring type microphone sound

Image shown on HMD

Figure 2 Overall view of the system

relatively easy to detect circles (and ellipses) within the real environment, so we have used an NTT DOCOMO logo as our marker, as shown in Fig. 3 (a). Any arbitrary object can be identified for use as the display surface by attaching this marker to it, and the position and orientation of the object can be determined by detecting the position and orientation of the marker.

To detect the marker, regions of color similar to the marker image are first detected, and after converting to a binary image, marker candidate regions are selected based on how closely they approximate an ellipse (Fig. 3 (b)). At this point, if multiple elliptical regions are detected, the largest such region in
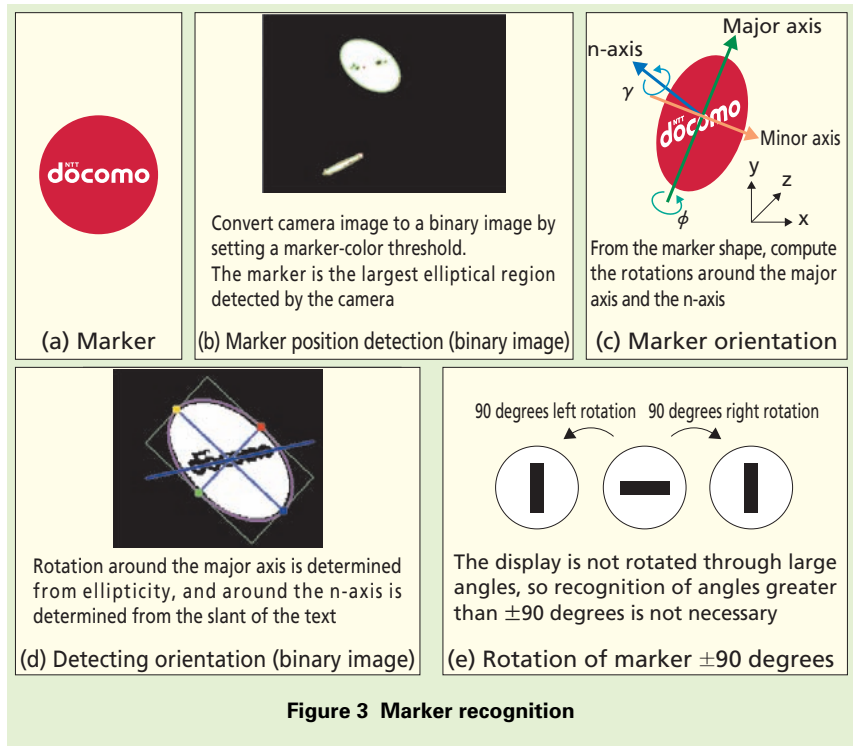

(a) Marker

Convert camera image to a binary image by setting a marker-color threshold.
The marker is the largest elliptical region detected by the camera

(b) Marker position detection (binary image)

From the marker shape, compute the rotations around the major axis and the n-axis

(c) Marker orientation

Rotation around the major axis is determined from ellipticity, and around the n-axis is determined from the slant of the text

(d) Detecting orientation (binary image)

90 degrees left rotation   90 degrees right rotation

The display is not rotated through large angles, so recognition of angles greater than ±90 degrees is not necessary

(e) Rotation of marker ±90 degrees

Figure 3 Marker recognition

the camera image is selected. For the orientation of the marker, the angle of rotation, ϕ, around the major axis and the angle of rotation, γ, around the normal vector, n-axis, through the center of the circle, must be calculated as shown in Fig. 3 (c).

The angle of rotation about the major axis can be calculated from the major and minor axes by equation (1).

$$\phi = \sin^{-1} (\text{minor axis length/major axis length}) \tag{1}$$

Actually, only the angle of rotation of the marker can be computed, and not the direction of rotation. Thus, we make the assumption that illumination is from above (positive y-axis), and by looking at the change in brightness over the elliptical region, we decide the direction of rotation around the major axis. Since we cannot fix the direction of rotation around the y-axis (when the y-axis is the major axis) with this method, we display the overlay content image in a rectangle proportional to the ratio of the major and minor axis lengths.

To compute the angle of rotation, γ, around the n-axis, we use the slant of the character region within the marker, as shown in Fig. 3 (d). As shown in Fig. 3 (e), using the slant of the character region it is only possible to detect angles of up to ±90 degrees from the initial position. However, considering the environmental conditions where this display device will be used, the display

will not be rotated through large angles, so we compute the angles based on the assumption that angles greater than ±90 degrees need not be recognized.

2) Detecting Object Location and Image Overlay

To detect the object region, we use color information in the image surrounding the marker. For our system we assumed that the marker would be placed at the top of the selected object, so we detect the color below the recognized marker. Next, we extract the object display area by finding a region approximating a rectangle of the detected color. The content image can be overlaid on the detected region of the selected object to use it as the display

(**Figure 4** (a)). By selecting a different object, the content can be displayed according to the size of the object as shown in Fig. 4 (b).

3) Detecting User Input Operations

User input operations are handled using the camera image from the HMD and a finger-ring type microphone. In the current system, we simplified operations by limiting them to touching the screen using the index finger.

First, the hand region is detected by finding a region within the selected-object region that is a different color than the selected object. When using the index finger to perform input on the content, only the index finger projects from the hand region, so we used the
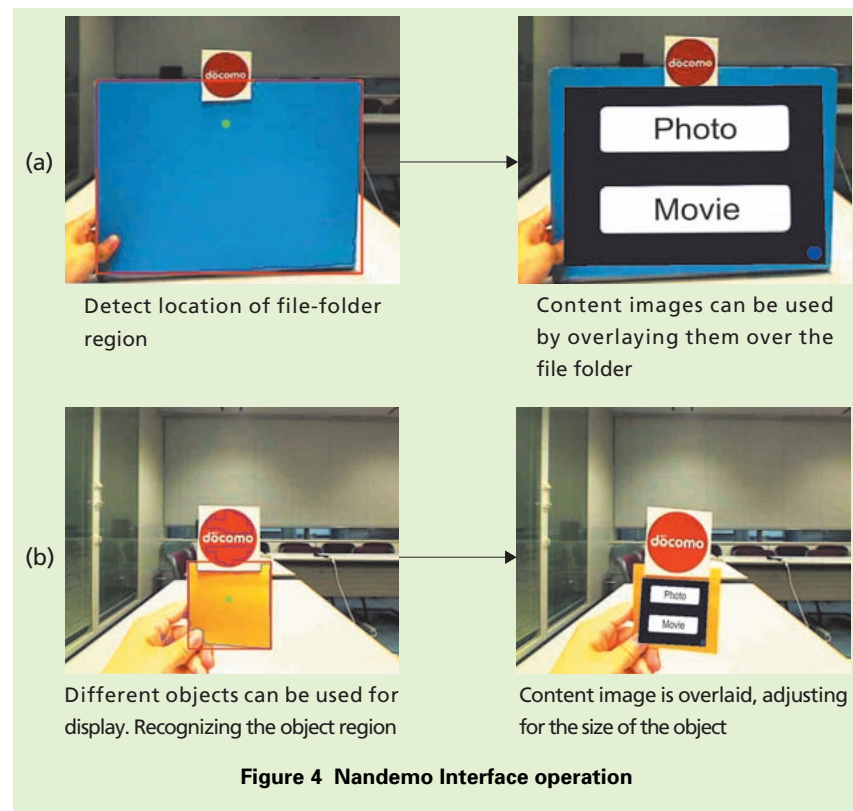

(a) Detect location of file-folder region

Content images can be used by overlaying them over the file folder

(b) Different objects can be used for display. Recognizing the object region

Content image is overlaid, adjusting for the size of the object

Figure 4 Nandemo Interface operation

uppermost point on the hand region as the touch location.

However, with this method, the hand holding the display object appears within the display object region in addition to the hand being used for the touch operation, resulting in recognition of multiple hand regions. We resolved this issue by observing the time sequence of changes in each region and selecting the region with more position fluctuations as the hand region.

To perform touch operations, the time of the touch must be detected in addition to recognizing the position of the user's finger. To do this, we use a microphone in the form of a finger ring, and by detecting the sound of the touch (of tapping the object), we detect the time (**Figure 5**). This allows, for example, tapping on a photo with a finger to select the desired photo, so the object can be used as a touch panel display.

4. Nandemo Interface User Survey

We performed a survey with six subjects regarding the effects on usability of using or not using physical objects. We used a file folder as the display object, and examined differences in usability when users held the folder and when they did not. For cases when not holding the folder, users performed input operations by moving their finger in the air rather than tapping the folder.

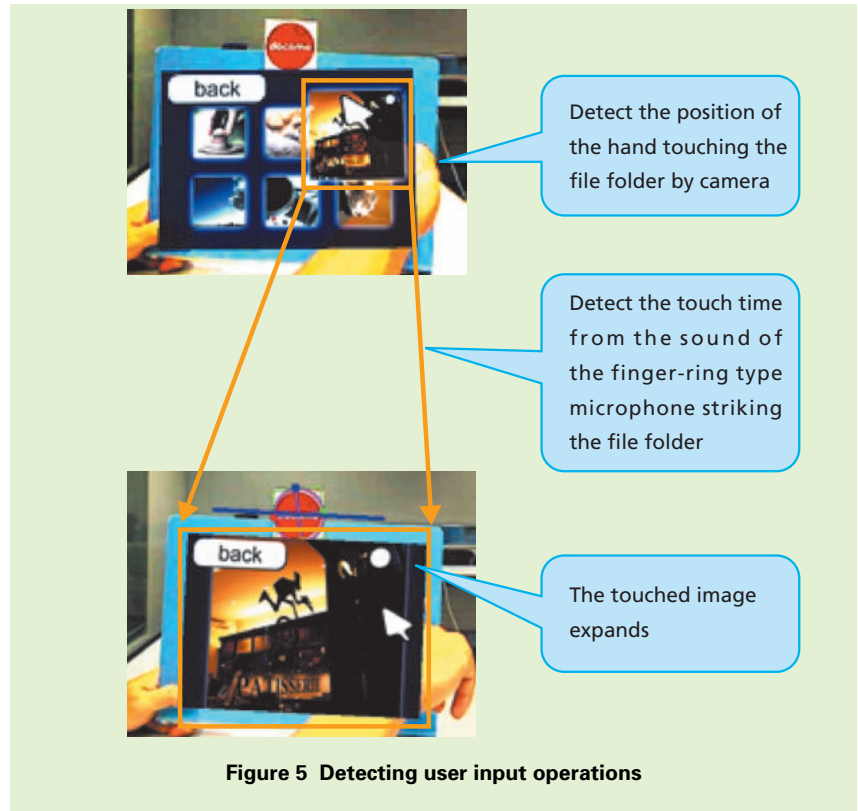We first evaluated the usability, comparing cases with and without a file

Detect the position of the hand touching the file folder by camera

Detect the touch time from the sound of the finger-ring type microphone striking the file folder

The touched image expands

Figure 5 Detecting user input operations

folder on a three-level scale of (1) Easier, (2) No difference and (3) More difficult. Four of the subjects responded that using a file folder was easier. The other two subjects answered that it was more difficult because holding the folder was tiring, and that motion of the content image as the hand moved made viewing difficult.

We also asked subjects to evaluate two aspects of the usability of input operations with the file folder on a three-level scale.

For the first aspect, we asked subjects to evaluate the ease in knowing the position of a button when using a file folder, compared to not using a folder, as (1) Easier, (2) No different or

(3) More difficult. Five subjects answered that it was easier and the remaining subject answered that it was no different.

For the second aspect, subjects evaluated whether there was a sense of having pushed a button as one of (1) Yes, there is a sense of pushing a button, (2) No different than without a file folder and (3) No sense of pushing a button. All subjects responded that there was a sense of pushing a button with the file folder.

In the survey results, five out of six subjects responded that the button location was easier to grasp when using a file folder. A possible reason for this is that with our system, users are always

holding the folder in the other hand, and they sense the position with their own hands in addition to using their eyes. With the HMD used in the current system, the same image is displayed to both eyes, so it is not possible to sense depth. In such an environment, the sense of holding the file folder plays an important role in getting a sense of distances, and this seems to be one way that the file folder contributes to knowing button locations more easily.

All subjects also indicated a stronger sense of pushing a button with the file folder. We used the sound of tapping the folder to detect input, so we suppose that the pressure feedback felt when tapping the folder had the additional effect contributing to this sense of pushing a button.

These results suggest that having a physical object is effective for interaction, but some subjects found using the folder more difficult. However, in these cases the reasons given included fatigue from holding the object and display shaking, so we can expect to improve on these difficulties by selecting the display object according to the situation. For example, a fixed object such as a desk or wall could be selected instead of a hand-held object when viewing over longer periods of time.

5. Conclusion

We have proposed a "Nandemo Interface" that uses nearby physical objects for displaying content and for input operations. With this system, the user is able to select a nearby object of size suitable for the content, and use it to view and operate on the content at the desired size. The system was evaluated with user surveys that indicated a clear increase in usability and ease of input operations when viewing content overlaid on a physical object.

The current system can only detect objects of a single color, but we plan to continue research to allow use of multicolored objects and to use objects without requiring the marker.

REFERENCES
[1] "15th Mobile Phone 'Personal Use' Survey 2009," Nikkei BP Consulting, July 2009 (In Japanese).
[2] S. Feiner, B. MacIntyre, M. Haupt and E. Solomon: "Windows on the World: 2D windows for 3D Augmented Reality," Proc. UIST '93, pp. 145-155, Nov. 1993.
[3] P. Mistry, P. Maes and L. Chang: "WUW - Wear Ur World - A Wearable Gestural Interface," Proc. CHI-9, ACM Press, pp. 4111-4116, 2009.
[4] M. Nakanishi and T. Horikoshi: "Effects of physical object's presence cues in the use of optical see-through HMDs," Technical report of IEICE —HIP, Vol. 109 (83), pp. 33-38, Jun. 2009 (In Japanese).