Pergamon

Comput. & Graphics, Vol. 18, No. 5, pp. 633–642, 1994

Printed in Great Britain
0097-8493/94 $6.00 + .00

0097-8493(94)00074-3

"FINGER-POINTER": POINTING INTERFACE BY IMAGE PROCESSING

MASAAKI FUKUMOTO, YASUHITO SUENAGA and KENJI MASE
NTT Human Interface Laboratories, Yokosuka, 238-03 Japan

Abstract—We have developed an experimental system for the 3D direct pointing interface "Finger-Pointer," which can recognize finger pointing actions and simple hand forms in real-time by processing the image sequences captured by stereoscopic TV cameras. The operator is not required to attach any peculiar device such as the Data-Glove or a magnetic sensor. Simple and fast image processing algorithms employed in the system enable real-time processing without any special image processing hardware. By introducing the notion of "VPO (Virtual Projection Origin)," the system can recognize pointing actions stably and accurately regardless of the operator's pointing style. The system also synchronizes and integrates the audio and the visual channels by introducing the "Timing Tag" technique.

1. INTRODUCTION

We think that there are two types of man-machine interface. One is the "professional" interface, which pursues high interaction speed, and the other is the "common" interface, which doesn't require any practice for use. The professional interface must be able to transmit the operator's intention to a machine rapidly and accurately, however, the necessity of practice or nuisance of setup is a trifling problem for this kind of interface. On the other hand, the common interface must be used by all people simply and easily. Accordingly, the practice needed to operate this interface must be reduced as much as possible. The keyboard, the most prevalent computer interface, is useful for command and character input, but it requires the user to practice a lot if he is to become fluent. We think information systems that will be used by everyone, such as the general information terminal supported by intelligent computer agents, must employ a common interface instead of professional interfaces such as keyboards. Especially, the common interfaces that adopt the interaction style used in ordinary human-to-human conversation don't require any practice to use, and everyone can operate that interface easily.

Human-to-Human interaction is composed of verbal and nonverbal modes[1]. The role of the nonverbal-mode, which encompasses posture, gesture, gaze, facial expression and so on, is as important as that of the verbal-mode. We have proposed the interface concept named "Human Reader"[2], which integrates verbal and nonverbal interaction modes by mainly using image processing techniques. A Human Reader consists of several recognition modules. "Head Reader"[3] recognizes human head motion such as brief responses (Yes/No). The "Face Reader"[4] understands human facial actions. In this paper, we focus on human-gestures that are extremely expressive in many nonverbal modes and use it to construct a human-computer interface.

Human gestures can be classified into three groups (Table 1). The first gesture group contains the pointing actions used to indicate 2D or 3D location; we call this the "locator" group. The next group, which we call "switcher," includes gestures that select between two or more states. The next group, named "valuator," contains gestures that indicate quantity (ex: "about *this* size" or "rotate *this* much"). The last group comprises gestures that visualize shapes, actions or feelings such as "*triangle*" or "*running*," we call this group "imager." Sign languages and body languages belong in this group. This imager group has strong expressive power, but the difficulty of accurate recognition is correspondingly increased.

Several interface prototype systems have been proposed. The multi modal graphics interface[5] employs "locator" gestures for pointing input. The finger-based commands used in many virtual reality systems[6] and the finger spelling[7] used by blind people belong in the "switcher" group. Some object manipulation gestures used in computer aided design tools[8] belong in the "valuator" group. The few systems recognize sign language gestures[9, 10], which have a relatively strict grammar, belong to the "imager" group.

In these systems, however, the operator is forced to wear special devices, such as Data-Glove or a magnetic-sensor. Some approaches using image processing methods free the operator from these devices. For example, the object handling system[11] recognizes pointing direction and hand forms by stereo TV cameras mounted above and in front of the operator, and the visual interface system[12] recognizes hand signs by a single TV camera in front of the operator. These systems, however, cannot work in real-time or need special image processing hardware.

Human beings normally interact by using plural communication modes simultaneously such as voice and gesture. The synchronization and integration of parallel input modes are necessary for realizing a multimodal computer interface. Some prototype systems using pointing and hand gestures[6,13] accept multimodal input messages such as a combination of pointing gestures and voice commands. In these systems,

Table 1. Classification of gesture.

Class	Content	Example
Locator	Indicate location in space	Pointing
Switcher	Select from some states	Hand spelling
Valuator	Indicate extents	Object manipulation
Imager	Indicate general images	Sign language, body language

however, the problem of synchronizing the input modules, all of which have different recognition speeds, has not been solved.

As the prototype of a gesture interface, we developed the human-pointing action recognition system called "Finger-Pointer." Gestures used in this system belong to the "locator," "switcher" and some "valuator" groups. By using a simple and fast image processing method, the system can recognize 3D pointing actions and simple hand forms in real-time without forcing the user to wear any special device. The operator can interact with the system by combination of pointing gestures and voice commands without concern for the time lag of each input channel. The next section outlines system construction of the "Finger-Pointer." Fast image processing methods for hand image detection are then described. Next, the new pointing direction determination method called "Virtual Projection Origin (VPO)" is proposed. Experiments have shown that VPO is very effective in various situations. Next, multichannel synchronization using "Timing-Tags" is described. Finally, remaining problems and possible applications of this system are discussed.

2. FINGER-POINTER

2.1. *System concept*

A main purpose of the Finger-Pointer system is to allow the user to communicate with various machines such as presentation or audio-visual instruments with pointing actions, hand forms and voice commands in a meeting space or a living room. Figure 1 shows the concept of the Finger-Pointer system. The operator's pointing actions are captured by two stereoscopic TV cameras: one mounted on the wall and the other on the ceiling. The system determines the 3D coordinates of the operator's finger tip by analyzing the camera images and uses the pointing direction as "locator." The system also can recognize several simple hand forms as "switcher" or "valuator" by analyzing the image captured by the camera mounted on the wall. The operator can communicate with the system using a natural combination of voice and gestures. In order to achieve more accurate pointing, the system can display a pointing cursor on the front screen to provide feedback to the user.

2.2. *Structure of the system*

Figure 2 shows a block diagram of the system. This system employs two monochrome CCD cameras driven by one "sync" signal. The ceiling camera image is converted into the "R" plane of the digitizing unit, and the wall camera image is converted into the "G" plane. These camera images are then digitized by the video digitizing unit of the graphic work station (GWS†). The operator's voice level is also digitized in the "B" plane and used to generate "Timing-Tags" (described later) for voice and gesture synchronization.

The system works on the GWS and processes 10 frames per second without any special image processing hardware. The user-specified, separated-word-type voice recognition unit (Voice Navigator) on a personal computer (Macintosh IIfx) is used for voice command recognition. By using a telescopic type microphone, the operator doesn't need to wear even a headset. Another GWS (personal IRIS) and a Hi-scanned Video Projector are used as the application platform.

† IRIS-4D/220GTX.

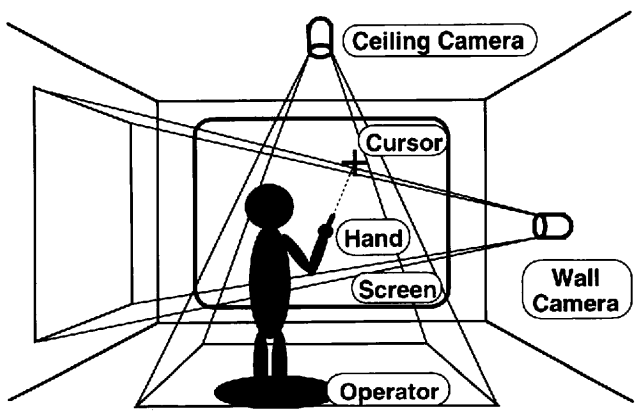

Fig. 1. Concept of the Finger-Pointer system. The operator's pointing actions are captured by two TV cameras, and the determined target is displayed on the screen.

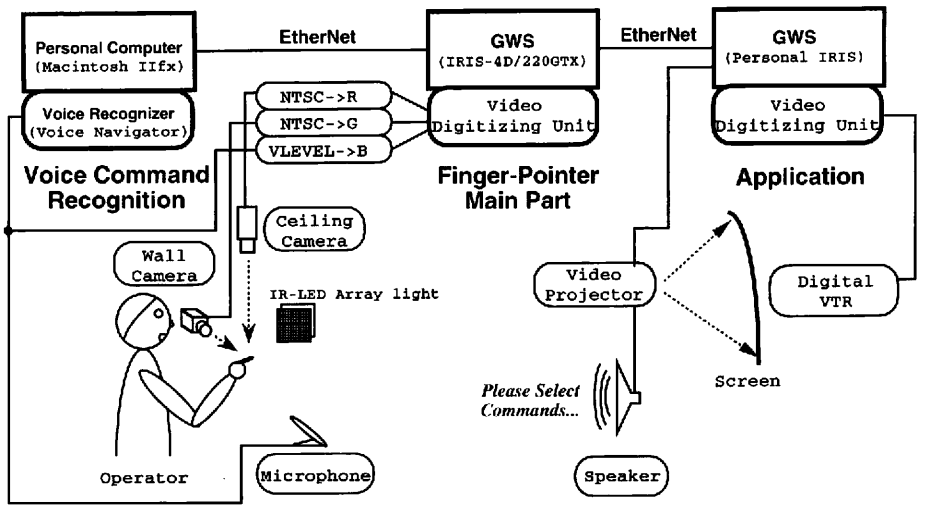

Fig. 2. Block diagram. Images of two TV cameras are converted into 'R' and 'G' plane, and digitized by GWS simultaneously.

3. IMAGE PROCESSING METHODS

3.1. *Finger-tip detection for "Locator"*

Figure 3 illustrates the method of determining finger tip location, and it's algorithm is described below.

1. Binarize the two images captured by the ceiling and the wall cameras with a fixed threshold, and extract hand regions.
2. Scan each binary image and determine the pixel that is closest to the screen, as the most likely candidate for the finger tip.
3. Calculate 3D position of the candidate pixel from the location of the candidate pixel in each image and camera parameters.
4. Decide whether that candidate pixel represents the real finger tip, based upon the length and thickness of the extracted region. (Length and thickness of index fingers and camera parameters are predetermined values.)

The system makes the following assumptions to assist finger-tip detection:

- When using a reflected light source, the hand region is lighter than the background region (for backlighting, it's darker).
- The operator's finger tip is the part of his body nearest the screen while he is pointing.
- The length and thickness of human fingers are not drastically different (no calibration is necessary for each operator).

Furthermore, the position at which the finger will next appear is estimated by the two most recent finger tip positions for speedy processing. If the operator's finger tip is included in the tracking area (about 8% of the captured image), the system can detect the finger tip candidate pixel quickly. After finger tip location is detected, the system determines the pointing direction by the use of the "Virtual Projection Origin" (described later).

3.2. *Thumb-click detection for "Trigger"*

Some trigger action is necessary to extracting a specific pointing direction because the pointing direction is continuously detected. Conceivable trigger actions are (a) existence of finger or hand, (b) static finger

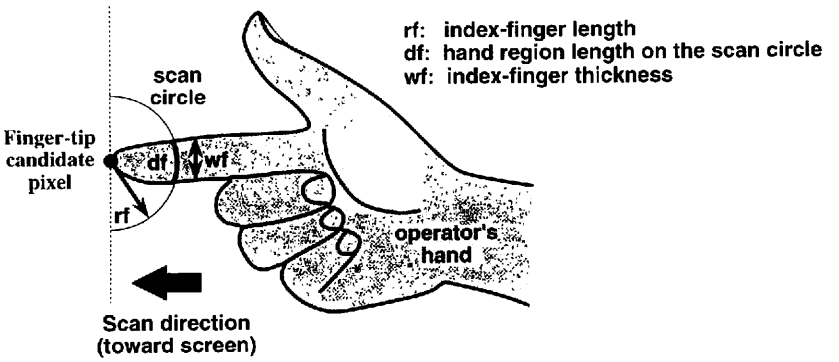

Fig. 3. Determining finger tip location. Scan the finger tip candidate that is closest to the screen, and confirm it based upon the length and thickness of the finger.

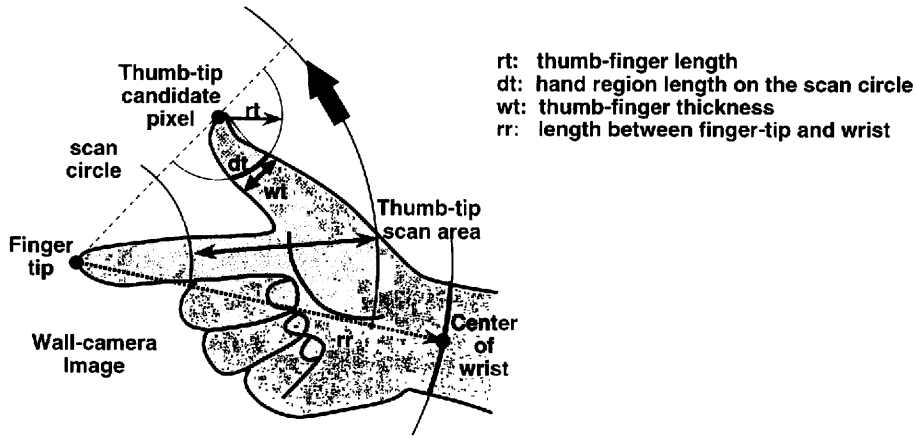

Fig. 4. Thumb-Switch detection. Scan the thumb tip candidate in the spreading fan manner from the line determined by the finger tip and wrist position.

position in some predetermined period, (c) the use of particular finger tip motion (example: drawing a small circle). All three of these actions reduce the pointing accuracy and operating speed. In addition, action classification (separation of "trigger" from "locator") is necessary. It is desirable that the "trigger" action become independent of the "locator" action, if possible. The Finger-Pointer system employs the thumb bending action as "trigger" and index finger direction as "locator," these two actions are basically independent. With appropriate system functions, the operator can use a click and drag function, similar to that possible with a one-button mouse.

The thumb scanning method is similar to the method used to detect the finger tip (Fig. 4).

1. Scan the binarized wall camera image and determine the wrist center.
2. Scan the image in a spreading fan pattern from the line determined by the operator's finger tip and wrist position, and determine the uppermost pixel of the hand region in the scanned area as the candidate for the thumb tip.
3. Decide whether that candidate pixel represents the real thumb tip, based upon the length and thickness of the extracted region. (Thumb length and thickness are predetermined values.)

3.3. *Finger-number detection for "Valuator"*

People often use their fingers to indicate numbers. We call this gesture "Finger-Number." The Finger-Pointer system can also recognize the number of outstretched fingers. The operator can communicate with the system by displaying different numbers of fingers. This feature allows more speedy selection than pointing to icons.

The recognition sequence is shown in Fig. 5.

1. Determine scan circle center from the finger tip and wrist center of the binarized wall camera image.
2. Sweep the scan circle and separate finger regions on the circle.
3. Decide how many fingers occupy each extracted region.

When using a low resolution camera, the boundary of neighboring fingers becomes indistinct. Thus recognizing finger numbers by counting isolated fingers would be inaccurate. The proposed method detects the correct number of fingers, even if two or more fingers are held together.

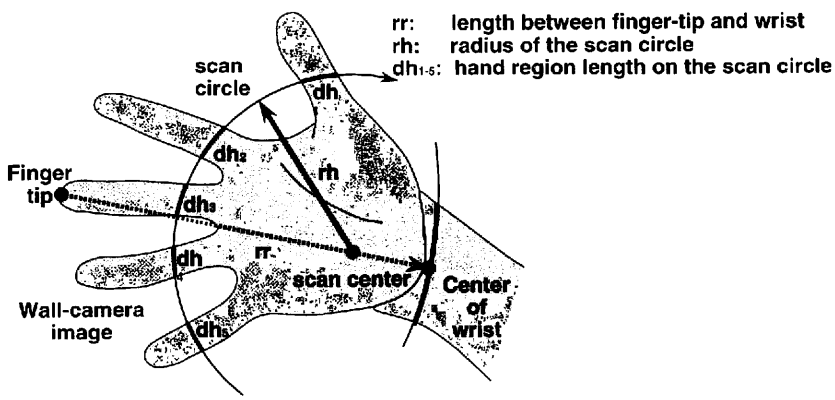

Fig. 5. Finger-Number detection. Sweep the scan circle and separate finger regions on the circle. Decide how many fingers occupy each extracted region.

3.4. *Lighting by infrared LED*

The Finger-Pointer system employs the fixed threshold binarization of camera images for achieving real-time processing. However, the performance of binarization with a fixed threshold is influenced by lighting conditions. A strong visible light is effective for stable binarization, but such a light makes the operator feel uncomfortable. This problem can be solved by using arrays of infrared LEDs as light sources; their light is unnoticeable to the operator. Filters that eliminate the visible spectrum are positioned in front of the CCD cameras. With this combination of infrared LEDs and filters, the system can achieve stable binarization with a fixed threshold regardless of the lighting condition of the room.

4. "VPO": VIRTUAL PROJECTION ORIGIN

4.1. *Pointing direction*

The operator's pointing direction is determined by a straight line that is defined by two points in 3-D space. We call these two points "Tip-Point" and "Base-Point." The Tip-Point corresponds to the operator's finger tip. However, the question is the location of the Base-Point. A preliminary experiment indicated that the position of the Base-Point is different for each operator. Even for the same operator, this point changes depending on the pointing style, for example, whether the user is tense or relaxed.

4.2. *Virtual projection origin*

For the Finger-Pointer system we employed a virtual Base-Point estimated by simple calibration before interaction. Therefore, the system can unify the user's desired pointing direction and the direction perceived by the system. We assume that the lines of pointing direction converge at one (Base) point when the operator points at several objects on a distant screen (Fig. 6). After this calibration, the operator's pointing direction can be expressed as the projection from the converged point through the operator's finger tip (Fig. 7). We call this point the "VPO"—Virtual Projection Origin.

The VPO calibration procedure is shown in Fig. 6.

1. Display predetermined marks on upper-right corner of the front screen.

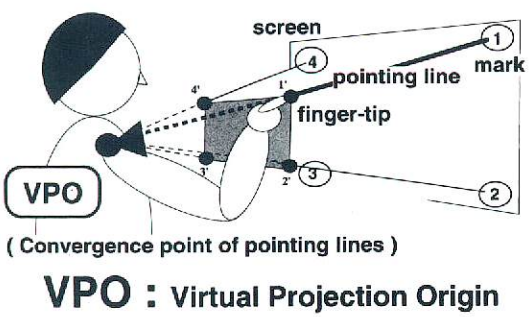

Fig. 6. VPO calibration. Estimate the convergence point (VPO) of pointing lines passing from the displayed marks through the corresponding finger tip position.

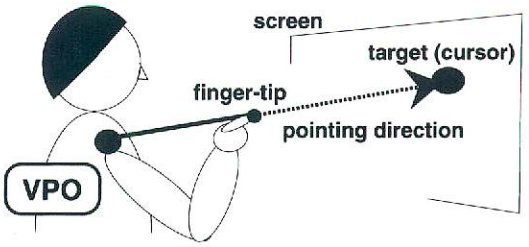

Fig. 7. Pointing by VPO. The operator's pointing direction is determined as a projection from the VPO through his finger tip.

2. Measure the operator's finger tip position when the operator points at the mark, and determine pointing line from the mark to the finger tip position.
3. Repeat this procedure and determine pointing lines for several other positions.
4. Estimate VPO as the point at which these pointing lines converge.

The VPO is the center of a sphere of minimum radius that is intersected by all pointing lines (Fig. 8). The sphere's radius indicates the convergence rate, and a small radius means good convergence (and thus accurate estimation).

4.3. *Distribution of VPO*

Figure 9 illustrates experimentally measured VPO distributions. The experiment tested 20 operators, and the distance between the operator and the 120-inch diagonal screen was 5.4 m. Each filled circle indicates the VPO position for one operator. The radius of each circle indicates the minimum sphere size intersected by all pointing lines (a small circle means good convergence). The figure shows that VPO position differs for each operator, and even for the same operator, the VPO position changes with the pointing style.

The experiment provided that the VPO of each operator converges within a 3.5-cm radius with a probability of 95%. By using the VPO method, the system has a pointing accuracy of 2.0° without cursor feedback, and 0.6° with cursor feedback, for all operators and pointing styles.

5. CHANNEL SYNCHRONIZATION BY "TIMING-TAG"

5.1. *Multi-modal pointing*

Human beings normally point using voice and gestures simultaneously. In this case, the finger is used for a "locator," and the voice is used for a "trigger." The combination of finger and voice provides more natural interaction than just hand gestures. For example, using voice allows the operator's hand to be raised just briefly for pointing so fatigue is less than occurs with thumb-switch triggering.

The Finger-Pointer provides integration of voice and pointing gesture by the use of a (user-specified, separated word type) speech recognition unit. The system integrates voice commands, pointing targets, thumb triggers, and finger numbers, and decides actions for

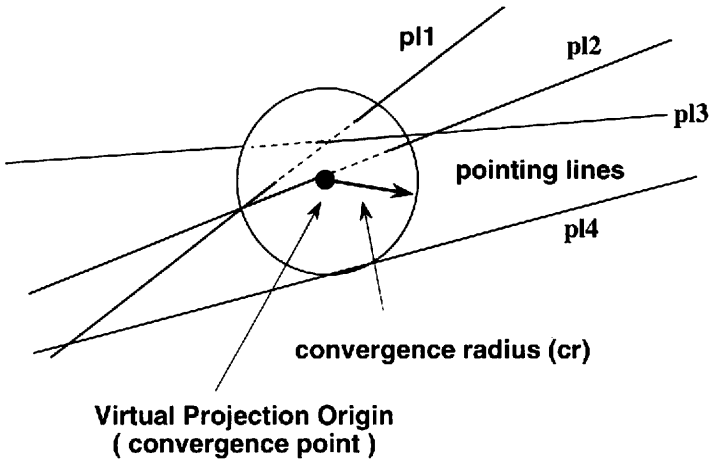

Fig. 8. Estimating convergence point. The VPO is estimated as the center of a sphere that has minimum radius and is intersected by all pointing lines.

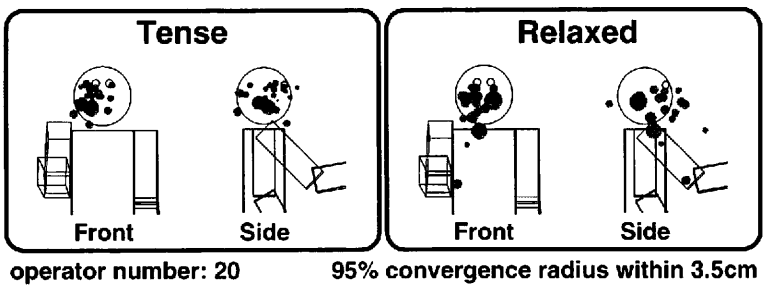

Fig. 9. VPO distribution. Each filled circle indicates the VPO position for one operator. The radius of each circle indicates convergence rate (small circle means good convergence).

the target. A typical combination example for a presentation application is shown in Table 2.

5.2. *Symbol mismatch*

In the Finger-Pointer system, pointing recognition can be completed in real-time, but the voice recognition unit needs a delay of about 0.5 seconds after the voice command is vocalized, and the delay changes depending on the size of the word dictionary and the word input. In general, inter-channel synchronization is necessary to integrate plural input channels. Each input channel has its own recognition module, and each module has a different recognition delay. Moreover, in many recognition modules, these delay times change depending on the input signals. If a group of events that take place at the same time are individually captured by each channel, the outputs of the recognition modules are randomly offset against each other. Therefore, the message integrator can't decide which symbols must be combined (Fig. 10).

5.3. *Timing-Tags*

The Finger-Pointer system realizes inter-channel synchronization by introducing "Timing-Tags." Figure 11 shows the concept of the Timing-Tag; it's algorithm is described below.

- Each recognition module and message integration module are driven by a master clock.
- When a recognition module receives a pointing event, it identifies the event with the current clock value, called the "Timing-Tag."
- After the recognition process, recognized symbols are passed to the integrator together with their corresponding timing-tags.
- The message integration module rearranges the recognized symbols according to their timing-tags.

Through the use of timing-tags, the processing delay of each recognition module can be neglected, and the system can quickly integrate multi-channel pointing

Table 2. Combination of gesture and voice commands for a presentation.

Pointing	Command (trigger, switcher)	Voice command sample
Index Finger	Thumb click	—
Index Finger	Voice	"This," "that," "from ," "to"
—	Voice + finger number	"Forward this many," "Back this many"
—	Voice	"Clear screen," "calibration"

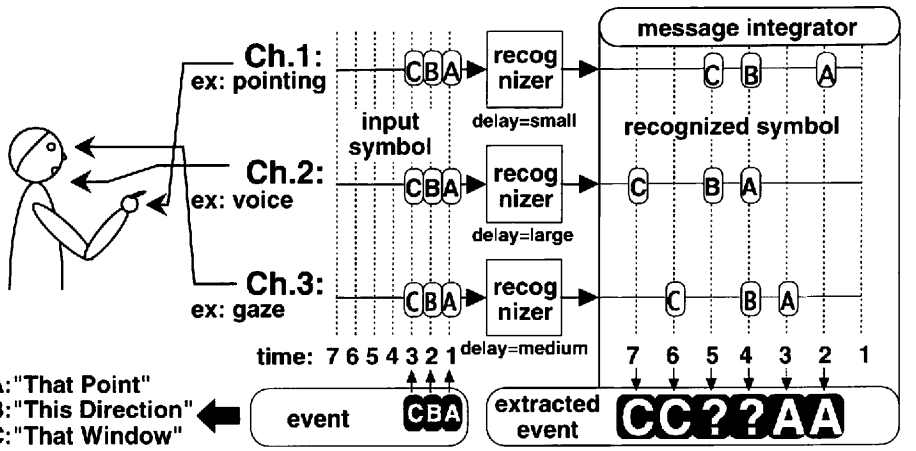

Fig. 10. Symbol mismatch. The message integrator cannot decide which symbols must be combined because of the different recognition delays of each module.

messages. Conventional voice recognizers, however, can't generate timing-tags. Thus, the timing-tag method is implemented in the system as a backsearching voice level mechanism (Fig. 12). When the recognized symbol is output, the system backsearches the recorded voice level buffer and finds the start and end times of the corresponding voice command. The timing-tag is generated from these times.

5.4. *Target detection by pointing velocity*

When using voice commands as a trigger, it is necessary to extract a specified pointing direction from several pointing directions measured while voice command is spoken. We noticed that the finger tip momentarily halts when it points at the targets. Consequently, the system can extract the correct pointing direction by determining the minimum velocity of the finger tip adjacent to the vocalization of a voice command (Fig. 13). The processing speed of our prototype system is, unfortunately, cannot follow quick pointing actions. Therefore, the system estimates target locus by interpolation of pointing directions.

6. APPLICATIONS

We constructed three applications based on the Finger-Pointer system.

6.1. *Presentation system*

Figure 14 illustrates a presentation system that uses a computer-based slide projector. Rectangular regions at the bottom of the screen serve as command buttons, for example, NextPage, PrevPage, ClearScreen, *etc.* The operator can select commands and emphasize the slide image in real-time by adding marks and lines. The operator can control the system by using several combination of gesture and voice commands (Table 2).

6.2. *Video browser*

"Finger-Pointer" can also be used as a video browsing system (Fig. 15). The operator can use hand motions and thumb-switch actions to control a VCR, for example: Play, Stop, and some special search operations similar to those offered by a "Shuttle-Ring."

6.3. *"Space-Writer"*

The system can detect alphanumerics and graphic figures written in space by the operator (Fig. 16). The pen-up/down operation is controlled by the operator's thumb-switch.

7. CONCLUSION

In this paper, we have introduced the new pointing action recognition system called Finger-Pointer. It does

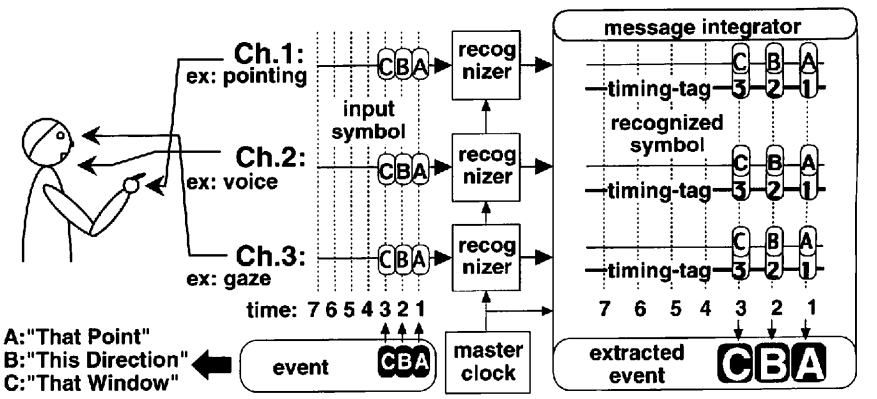

Fig. 11. Inter-channel synchronization by Timing-Tag. Occurrence time of each event is clarified by the use of Timing-Tags. Symbols can be combined correctly.

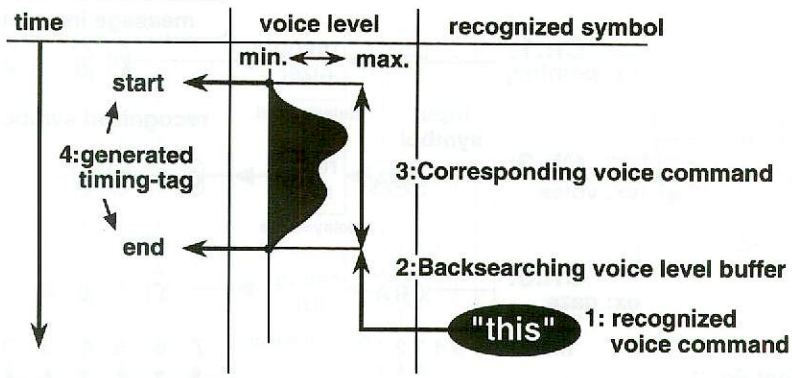

Fig. 12. Voice level backsearch. The system uses conventional voice recognition unit, and the Timing-Tag is synthesized based on vocalized voice level transition.

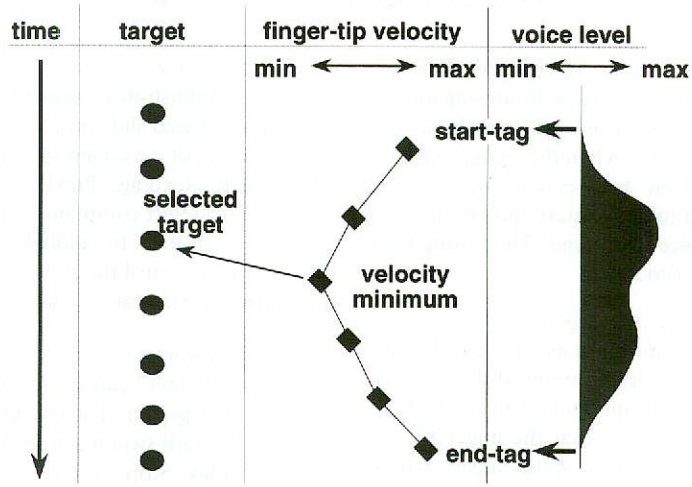

Fig. 13. Target estimation by velocity. When a voice command is vocalized, the system extracts the target indicated by minimum finger tip velocity.

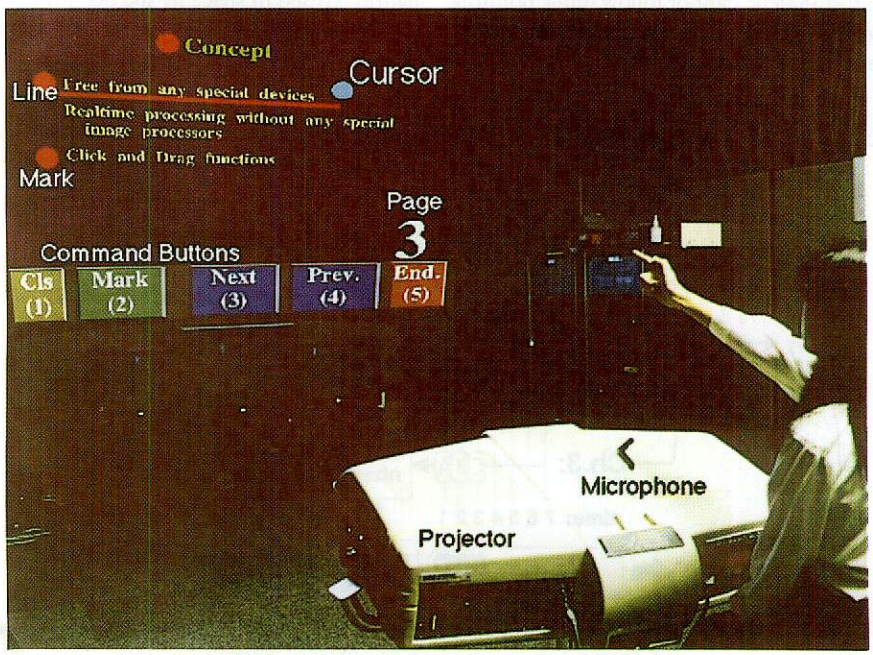

Fig. 14. Presentation system. The operator can control and emphasize the projected image using combinations

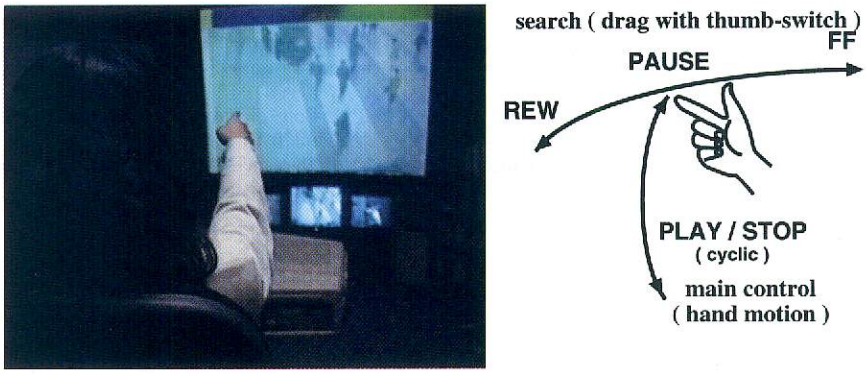

Fig. 15. Video browser. The operator can control a VCR by hand motions and thumb switching.

Fig. 16. Drawing characters in space. The system can detect letters and simple figures written in space by finger.

not force the operator to wear special devices and realize a more human-friendly interface. By introducing the notion of the VPO (Virtual Projection Origin), the system can detect stable and accurate pointing regardless of the operator's pointing style. The system also realizes multi-channel message synchronization by introducing Timing-Tags for integrating several recognition modules, all of which have different processing delays. The system operates in real-time without any special image processing hardware, so it is useful as the platform of a multi-modal interface for many applications such as presentation, navigation, machine control, and so on.

The current prototype system cannot cope with operator movement after calibration. But the notion of the VPO will be applicable for pointing with movement by tracking the operator with TV cameras or another sensors, and determining the VPO relative to the operator's location. Improvement of processing speed together with higher pointing accuracy and the recognition of more complex hand gestures still remain as future problems.

REFERENCES

1. M. Vargas, *Louder than words*, Iowa State University Press (1987).
2. Y. Suenaga, K. Mase, M. Fukumoto, and Y. Watanabe, Human reader: An advanced man-machine interface based on human images and speech. *Trans. IEICE*† **J75-D-II**, 190–202 (1987) (in Japanese).
3. K. Mase, Y. Watanabe, and Y. Suenaga, A real time head motion detection system. *Proc. SPIE* **1260**, 262–269 (1980).
4. K. Mase, An application of optical flow—Extraction of facial expression. *Proc. IAPR MVA* **'90**, 195–198 (1990).
5. R. A. Bolt, Put-that-there: Voice and gesture at the graphics interface. *ACM SIGGRAPH* **14**, 262–270 (1980).
6. D. Weimer and S. K. Ganapathy, A synthetic visual environment with hand gestures and voice input. *Proc. CHI* **'89**, 235–240 (1989).
7. K. Hirose, Recognition of Japanese manual alphabet using thinning image. *IEICE Tech. Rep.* **CV84-1**, 1–6 (1993) (in Japanese).
8. M. Kato, M. Fukino, T. Oyama, and M. Miwa, A gesture interface for 3D shape manipulation. *Proc. IEICE Fall Conf.* **A-127** (1991) (in Japanese).
9. C. Charayaphan and A. E. Marble, An image processing system for interpreting motion in american sign language. *Proc. Vision Interface* **'90**, 23–30 (1990).
10. S. Tamura and S. Kawasaki, Recognition system for sign language motion image. *IPSJ*‡ *SIG Notes* **86-CV-44-1**, 1–8 (1986).

† IEICE: The Institute of Electronics, Information and Communication Engineers of Japan

‡ IPSJ: Information Processing Society of Japan

11. K. Ishibuchi, H. Takemura, and F. Kishino, Real-time hand shape recognition using pipeline image processor. *SICE-HI*† *News & Rep.* 7, 275–280 (1992) (in Japanese).
12. W. Wongwarawipet and M. Ishizuka, A visual interface for transputer network (VIT) and its application to moving image analysis. *Proc. 3rd Transputer/Occam Int'l Conf.* (1990).
13. R. A. Bolt, The integrated multi-modal interface, *Trans. IEICE* **J70-D**, 2017–2025 (1987).
14. M. W. Krueger, *Artificial Reality*, Addison-Wesley Publishing Company, Reading, MA (1983).

† SICE: Society of Instrument and Control Engineers of Japan