Office de la Propriété Intellectuelle du Canada
Un organisme d'Industrie Canada

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2412878 C 2015/02/03

(11)(21) 2 412 878

(12) BREVET CANADIEN
CANADIAN PATENT

(13) C

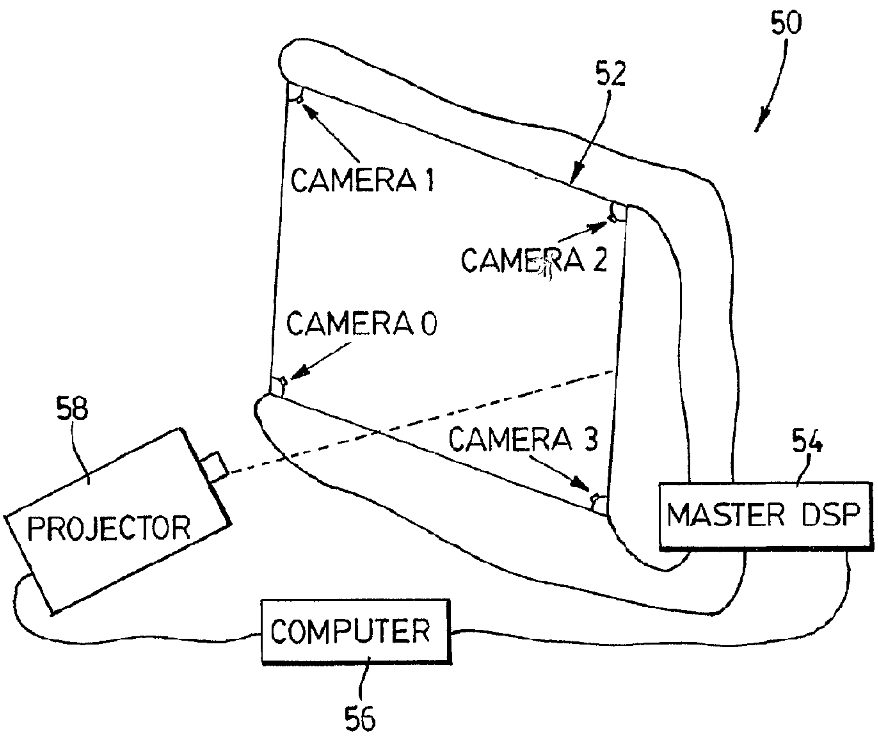
(86) Date de dépôt PCT/PCT Filing Date: 2001/07/05
(87) Date publication PCT/PCT Publication Date: 2002/01/10
(45) Date de délivrance/Issue Date: 2015/02/03
(85) Entrée phase nationale/National Entry: 2002/12/16
(86) N° demande PCT/PCT Application No.: CA 2001/000980
(87) N° publication PCT/PCT Publication No.: 2002/003316
(30) Priorités/Priorities: 2000/07/05 (US09/610,481); 2001/06/01 (US60/294,611)

(51) Cl.Int./Int.Cl. *G06F 3/042* (2006.01)
(72) Inventeurs/Inventors:
MORRISON, GERALD, CA;
HOLMGREN, DAVID, CA

(73) Propriétaire/Owner:
SMART TECHNOLOGIES ULC, CA

(74) Agent: SIM & MCBURNEY

(54) Titre : SYSTEME TACTILE A CAMERA
(54) Title: CAMERA-BASED TOUCH SYSTEM



(57) Abrégé/Abstract:
A camera-based touch system (50) includes a passive touch surface (60) and at least two cameras (63) associated with the touch surface. The at least two cameras (63) have overlapping fields of view (FOV) encompassing the touch surface. The at least two cameras (63) acquire images of the touch surface from different locations and generate image data. A processor (54) receives and processes image data generated by the at least two cameras to determine the location of the pointer relative to the touch surface when the pointer is captured in images acquired by the at least two cameras. Actual pointer contact with the touch surface and pointer hover above the touch surface can be determined.

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) **World Intellectual Property Organization**
International Bureau

(43) **International Publication Date**
10 January 2002 (10.01.2002)

PCT

(10) **International Publication Number**
WO 02/03316 A1

(51) **International Patent Classification**[7]: **G06K 11/08**

(21) **International Application Number:** PCT/CA01/00980

(22) **International Filing Date:** 5 July 2001 (05.07.2001)

(25) **Filing Language:** English

(26) **Publication Language:** English

(30) **Priority Data:**
09/610,481 5 July 2000 (05.07.2000) US
Not furnished 1 June 2001 (01.06.2001) US

(71) **Applicant** *(for all designated States except US)*: **SMART TECHNOLOGIES INC.** [CA/CA]; Suite 600, 1177 - 11th Avenue S.W., Calgary, Alberta T2R 1K9 (CA).

(72) **Inventors; and**
(75) **Inventors/Applicants** *(for US only)*: **MORRISON, Gerald** [CA/CA]; 736 Arbour Lake Drive NW, Calgary, Alberta T3G 4C6 (CA). **HOLMGREN, David** [CA/CA]; 1097 Cranston Drive SE, Calgary, Alberta T3M 4T6 (CA).

(74) **Agent: RUSTON, David, A.**; SIM & MCBURNEY, Sixth Floor, 330 University Avenue, Toronto, Ontario M5G 1R7 (CA).

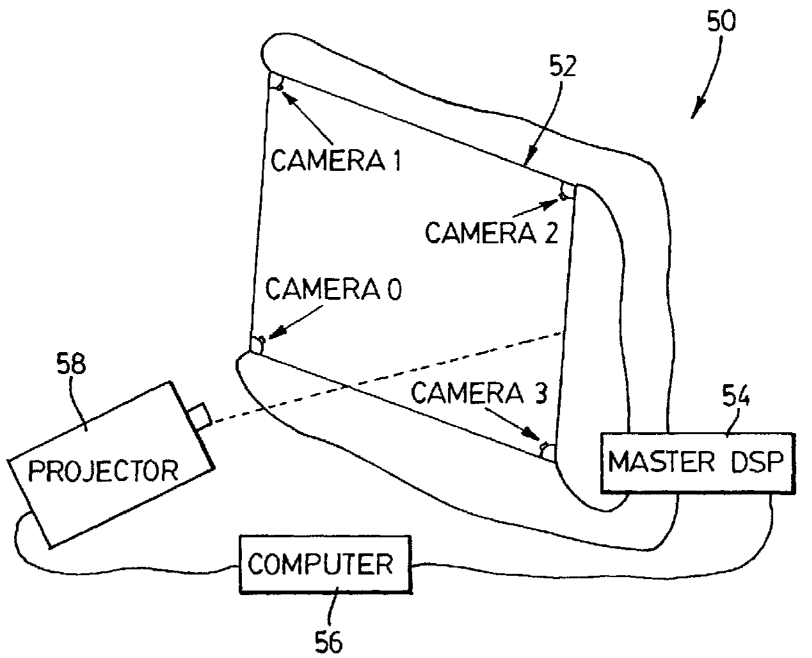
(81) **Designated States** *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) **Designated States** *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report*

[Continued on next page]

(54) **Title:** CAMERA-BASED TOUCH SYSTEM



(57) **Abstract:** A camera-based touch system (50) includes a passive touch surface (60) and at least two cameras (63) associated with the touch surface. The at least two cameras (63) have overlapping fields of view (FOV) encompassing the touch surface. The at least two cameras (63) acquire images of the touch surface from different locations and generate image data. A processor (54) receives and processes image data generated by the at least two cameras to determine the location of the pointer relative to the touch surface when the pointer is captured in images acquired by the at least two cameras. Actual pointer contact with the touch surface and pointer hover above the touch surface can be determined.

WO 02/03316 A1

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

CAMERA-BASED TOUCH SYSTEM

TECHNICAL FIELD

The present invention relates generally to touch systems and in particular to a camera-based touch system.

BACKGROUND ART

Touch systems are well known in the art and typically include a touch screen having a touch surface on which contacts are made using a pointer in order to generate user input. Pointer contacts with the touch surface are detected and are used to generate corresponding output depending on areas of the touch surface where the contacts are made. There are basically two general types of touch systems available and they can be broadly classified as "active" touch systems and "passive" touch systems.

Active touch systems allow a user to generate user input by contacting the touch surface with a special pointer that usually requires some form of on-board power source, typically batteries. The special pointer emits signals such as infrared light, visible light, ultrasonic frequencies, electromagnetic frequencies, etc. that activate the touch surface.

Passive touch systems allow a user to generate user input by contacting the touch surface with a passive pointer and do not require the use of a special pointer in order to activate the touch surface. A passive pointer can be a finger, a cylinder of some material, or any suitable object that can be used to contact some predetermined area of interest on the touch surface.

Passive touch systems provide advantages over active touch systems in that any suitable pointing device, including a user's finger, can be used as a pointer to contact the touch surface. As a result, user input can easily be generated. Also, since special active pointers are not necessary in passive touch systems, battery power levels and/or pointer damage, theft, or pointer misplacement are of no concern to users.

Passive touch systems have a number of applications relating to computer operation and video display. For example, in one interactive application, as is disclosed in U.S. Patent No. 5,448,263 to Martin, assigned to the assignee of the present invention, a passive touch system is coupled to a computer and the computer display is presented on the touch surface of the touch screen. The coordinates representing specific locations on the touch surface are mapped to the computer display. When a user contacts the touch surface, the coordinates of the contact position are fed back to the computer and mapped to the computer display thereby allowing the user to operate the computer in a manner similar to using a computer mouse simply by contacting the touch surface. Furthermore, the coordinates fed back to the computer can be recorded in an application and redisplayed at a later time. Recording contact coordinates is typically done when it is desired to record information written or drawn on the touch surface by the user.

The resolution of a passive touch screen determines if the touch system is suitable for recording information written or drawn on the touch screen or only useful for selecting areas on the touch screen mapped to regions on the computer or video display in order to manipulate the computer or video display. Resolution is typically measured in dots per inch (DPI). The DPI is related to the size of the touch screen and the sampling ability of the touch system hardware and software used to detect contacts on the touch surface.

Low-resolution passive touch screens only have enough DPI to detect contacts on the touch surface within a large group of pixels displayed by the computer or video display. Therefore, these low-resolution passive touch screens are useful only for manipulating the computer or video display.

On the other hand, high-resolution passive touch screens have sufficient DPI to detect contacts that are proportional to a small number of pixels or sub-pixels of the computer or video display. However, a requirement for high-resolution touch screens is the ability to detect when the pointer is in contact with the touch surface. This is necessary for writing, drawing, mouse-click operations, etc. Without the ability to detect pointer contact with the touch screen, writing and drawing would be one continuos operation, and mouse clicks would not be possible thereby making computer display manipulation virtually impossible. A secondary requirement is the ability to detect when the pointer is “hovering” above the touch surface. Although not required for writing or drawing, today’s computer operating systems are increasingly using hover information to manipulate computer or video displays or pop-up information boxes.

Passive touch screens are typically either of the analog resistive type, surface acoustic wave (SAW) type or capacitive type. Unfortunately, these touch screens suffer from a number of problems or shortcomings as will be described.

Analog resistive touch screens typically have a high-resolution. Depending on the complexity of the touch system, the resolution of the touch screen can produce 4096x4096 DPI or higher. Analog resistive touch screens are constructed using two flexible sheets that are coated with a resistive material and arranged as a sandwich. The sheets do not come into contact with each other until a contact has been made. The sheets are typically kept separated by insulating microdots or by an insulating air space. The sheets are constructed from ITO, which is mostly transparent. Thus, the touch screen introduces some image distortion but very little parallax.

During operation of an analog resistive passive touch screen, a uniform voltage gradient is applied in one direction along a first of the sheets. The second sheet measures the voltage along the first sheet when the two sheets contact one another as a result of a contact made on the touch surface. Since the voltage gradient of the first sheet can be translated to the distance along the first sheet, the measured voltage is proportional to the position of the contact on the touch surface. When a contact coordinate on the first sheet is acquired, the uniform voltage gradient is then applied to the second sheet and the first sheet measures the voltage along the second sheet. The voltage gradient of the second sheet is proportional to the distance along the second sheet. These two contact coordinates represent the X-Y position of the contact on the touch surface in a Cartesian coordinate system.

Unfortunately, since mechanical pressure is required to bring both sheets into contact, analog resistive touch screens can only detect contact when there is sufficient pressure to bring the two sheets together. Analog resistive passive touch screens also cannot sense when a pointer is hovering over the touch surface. Therefore, in the case of analog resistive touch screens contact events and positions can only be detected when actual contacts are made with the touch surface.

Surface acoustic wave (SAW) touch screens typically provide for medium resolution and are not suitable for recording good quality writing. SAW touch screens employ transducers on the borders of a glass surface to vibrate the glass and produce acoustic waves that ripple over the glass surface. When a contact is made on the glass surface, the acoustic waves reflect back and the contact position is determined from the signature of the reflected acoustic waves.

Unfortunately, SAW touch screens exhibit noticeable parallax due to the thickness of the vibrating glass that is placed over the surface of the video or computer display. Also, contact events and positions can only be detected when actual contacts are made with the glass surface. Furthermore, SAW touch screens do not scale beyond a few feet diagonal.

Capacitive touch screens provide for low resolution because contacts can only be determined in large areas (approximately ½"x ½"). As a result, capacitive touch screens cannot be used for recording writing or drawing but are suitable for selecting areas on the touch screen corresponding to computer generated buttons displayed on the video or computer display. Capacitive touch screens also suffer disadvantages in that they are sensitive to temperature and humidity. Similar to analog resistive touch screens and SAW touch screens, capacitive touch screens can also only detect contact events and positions when actual contacts are made with the touch surface.

Scalability of passive touch screens is important since the demand for larger electronic digitizers is increasing. Where digitizers were once small desktop appliances, today they have found there way onto electronic whiteboarding applications. The need to build a passive touch sensitive "wall" has become a requirement for new touch screen applications. Existing passive touch screens of the types discussed above are all limited in the maximum size where they are still functional.

As will be appreciated, improvements to passive touch systems are desired. It is therefore an object of the present invention to provide a novel camera-based touch system.

DISCLOSURE OF THE INVENTION

According to one aspect of the present invention there is provided a camera-based touch system comprising:

at least two cameras associated with a passive touch surface and having overlapping fields of view encompassing said touch surface, said at least two cameras acquiring images of said touch surface from different locations and generating image data; and

a processor receiving and processing image data generated by said at least two cameras to determine the location of a pointer relative to said touch surface when said pointer is captured in images acquired by said at least two cameras.

Preferably the at least two cameras are digital cameras having fields of view looking generally along the plane of the touch surface. The image data generated by each digital camera includes a pointer median line x and a pointer tip location z. Each of the digital cameras includes a pixel array having selectable pixel rows. Pixel intensities of pixels in the selectable pixel rows are used during generation of the image data. Preferably, pixel intensities of pixels in a region of interest within the selectable pixel rows are used during generation of the image data.

In a preferred embodiment, each of the digital cameras includes a CMOS image sensor and a digital signal processor. The digital signal processor receives image output from the image sensor and executes a find pointer routine to determine if a pointer is in each image acquired by the digital camera and if so, the median line of the pointer. It is also preferred that the digital signal processor of each digital camera executes an update background image routine to update the background image after each image is acquired. Preferably, the digital signal processor of each digital camera further determines the differences between each acquired image and the background image to detect changing light conditions.

According to another aspect of the present invention there is provided a camera-based touch system comprising:

a generally rectangular passive touch surface on which contacts are made using a pointer;

a digital camera mounted adjacent each corner of said touch surface, said digital cameras having overlapping fields of view encompassing said touch surface, said digital cameras acquiring images of said touch surface and generating image data that includes the median line x and pointer tip location z of a pointer when said pointer is captured in images acquired by said digital cameras; and a processor receiving and processing image data generated by said digital cameras to determine the location of said pointer relative to said touch surface and whether said pointer is in contact with said touch surface.

According to yet another aspect of the present invention there is provided a method of detecting the position of a pointer relative to a touch surface comprising the steps of:

acquiring images of said touch surface from different locations using cameras having overlapping fields of view and generating image data; and

processing said image data to detect the existence of a pointer within said acquired images and to determine the location of said pointer relative to said touch surface.

The present invention provides advantages in that the passive touch system is of high resolution and allows actual pointer contacts with the touch surface as well as pointer hovers above the touch surface to be detected and corresponding output generated. Also, the present passive touch system provides advantages in that it does not suffer from parallax, image distortion, pointer position restrictions, image projection and scalability problems that are associated with prior art passive touch systems.

Furthermore, the present invention provides advantages in that since CMOS digital cameras are used, arbitrary pixel rows in the digital camera pixel arrays can be selected. This enables the frame rates of the digital cameras to be increased significantly. Also, since the pixel rows can be arbitrary selected, the pixel arrays can be exposed for greater durations for given digital camera frame rates allowing for good operation in dark rooms as well as well lit rooms.

BRIEF DESCRIPTION OF THE DRAWINGS

Embodiments of the present invention will now be described more fully with reference to the accompanying drawings in which:

Figure 1 is a schematic diagram of a camera-based touch system in accordance with the present invention;

Figure 2 is an isometric view of a touch screen forming part of the touch system of Figure 1;

Figure 3 is an isometric view of a corner portion of the touch screen of Figure 2;

Figure 4 is a schematic diagram of a digital camera forming part of the touch screen of Figure 2;

Figure 5 is a schematic diagram of a master controller forming part of the touch system of Figure 1;

Figure 6 is a flowchart showing the steps performed during execution of a processFrame routine;

Figure 7 is a flowchart showing the steps performed during execution of a segmentPointer routine;

Figure 8 is a flowchart showing the steps performed during execution of a findPointer routine;

Figure 9 shows an image acquired by a digital camera and a pixel subset of the image that is processed;

Figure 10 shows a region of interest (ROI) within the pixel subset of Figure 9;

Figure 11 shows triangulation geometry used to calculate a pointer contact position on the touch surface of the touch screen illustrated in Figure 2;

Figure 12 shows an image acquired by a digital camera including the pointer tip and its median line;

Figure 13 shows pointer contact and pointer hover for different orientations of the pointer;

Figure 14 is an image of the touch surface of the touch screen as seen by a digital camera;

Figures 15 and 16 show the results of a Matlab simulation of pointer tracking using a Kalman filter; and

Figures 17a to 17d show the results of another Matlab simulation of pointer tracking using a Kalman filter.

BEST MODE FOR CARRYING OUT THE INVENTION

Turning now to Figure 1, a camera-based touch system in accordance with the present invention is shown and is generally identified by reference numeral

50. As can be seen, touch system 50 includes a touch screen 52 coupled to a digital signal processor (DSP) based master controller 54. Master controller 54 is also coupled to a computer 56. Computer 56 executes one or more application programs and provides display output that is presented on the touch screen 52 via a projector 58. The touch screen 52, master controller 54, computer 56 and projector 58 form a closed-loop so that user contacts with the touch screen 52 can be recorded as writing or drawing or used to control execution of application programs executed by the computer 56.

Figures 2 to 4 better illustrate the touch screen 52. Touch screen 52 includes a touch surface 60 bordered by a rectangular frame 62. Touch surface 60 is in the form of a rectangular planar sheet of passive material. DSP-based CMOS digital cameras 63 are associated with each corner of the touch screen 52. Each digital camera 63 is mounted on a frame assembly 64. Each frame assembly 64 includes an angled support plate 66 on which the digital camera 63 is mounted. Supporting frame elements 70 and 72 are mounted on the plate 66 by way of posts 74 and secure the plate 66 to the frame 62.

Each digital camera 63 includes a two-dimensional CMOS image sensor and associated lens assembly 80, a first-in-first-out (FIFO) buffer 82 coupled to the image sensor and lens assembly 80 by a data bus and a digital signal processor (DSP) 84 coupled to the FIFO 82 by a data bus and to the image sensor and lens assembly 80 by a control bus. A boot EPROM 86 and a power supply subsystem 88 are also included.

In the present embodiment, the CMOS camera image sensor is a Photobit PB300 image sensor configured for a 20x640 pixel subarray that can be operated to capture image frames at rates in excess of 200 frames per second since arbitrary pixel rows can be selected. Also, since the pixel rows can be arbitrarily selected, the pixel subarray can be exposed for a greater duration for a given digital camera frame rate allowing for good operation in dark rooms as well as well lit rooms. The FIFO buffer 82 is manufactured by Cypress under part number CY7C4211V and the DSP 84 is manufactured by Analog Devices under part number ADSP2185M.

The DSP 84 provides control information to the image sensor and lens assembly 80 via the control bus. The control information allows the DSP 84 to control parameters of the image sensor and lens assembly 80 such as exposure, gain, array configuration, reset and initialization. The DSP 84 also provides clock signals to the image sensor and lens assembly 80 to control the frame rate of the image sensor and lens assembly 80.

The angle of the plate 66 is selected so that the field of view (FOV) of each digital camera 63 extends beyond a designated peripheral edge of the touch surface 60 as shown in Figure 11. In this way, the entire touch surface 60 is within the fields of view of the digital cameras 63.

Master controller 54 is best illustrated in Figure 5 and includes a DSP 90, a boot EPROM 92, a serial line driver 94 and a power supply subsystem 95. The DSP 90 communicates with the DSPs 84 of the digital cameras 63 over a data bus via a serial port 96 and communicates with the computer 56 over a data bus via a serial port 98 and the serial line driver 94. In this embodiment, the DSP 90 is also manufactured by Analog Devices under part number ADSP2185M. The serial line driver 94 is manufactured by Analog Devices under part number ADM222.

The master controller 54 and each digital camera 63 follow a communication protocol that enables bi-directional communications via a common serial cable similar to a universal serial bus (USB). The transmission bandwidth is divided into thirty-two (32) 16-bit channels. Of the thirty-two channels, six (6) channels are assigned to each of the DSPs 84 in the digital cameras 63 and to the DSP 90 in the master controller 54 and the remaining two (2) channels are unused. The master controller 54 monitors the twenty-four (24) channels assigned to the DSPs 84 while the DSPs 84 monitor the six (6) channels assigned to the DSP 90 of the master controller 54. Communications between the master controller 54 and the digital cameras 63 are performed as background processes in response to interrupts.

The general operation of the touch system 50 will now be described. Each digital camera 63 acquires images of the touch surface 60 within the field of view of its image sensor and lens assembly 80 at a desired frame rate and processes each acquired image to determine if a pointer is in the acquired image. If a pointer is in the acquired image, the image is further processed to determine characteristics of the pointer contacting or hovering above the touch surface 60. Pointer information packets (PIPs) including pointer characteristics, status and/or diagnostic information are then generated by the digital cameras 63 and the PIPs are queued for transmission to the master controller 54. The digital cameras 63 also receive and respond to command PIPs generated by the master controller 54.

The master controller 54 polls the digital cameras 63 for PIPs. If the PIPs include pointer characteristic information, the master controller 54 triangulates pointer characteristics in the PIPs to determine the position of the pointer relative to the touch surface 60 in Cartesian rectangular coordinates. The master controller 54 in turn transmits calculated pointer position data, status and/or diagnostic information to the personal computer 56. In this manner, the pointer position data transmitted to the personal computer 56 can be recorded as writing or drawing or can be used to control execution of application programs executed by the computer 56. The computer 56 also updates the display output conveyed to the projector 58 so that information presented on the touch surface 60 reflects the pointer activity.

The master controller 54 also receives commands from the personal computer 56 and responds accordingly as well as generates and conveys command PIPs to the digital cameras 63.

Specifics concerning the processing of acquired images and the triangulation of pointer characteristics in PIPs will now be described with particular reference to Figures 6 to 8.

Initially, a camera offset angle calibration routine is performed to determine the offset angle δ of each digital camera 63 (see Figure 11) so that the contact or hover position of a pointer relative to the touch surface 60 can be accurately determined. Details of the camera offset angle calibration are described in U.S. Patent No. 6,919,880 entitled “Calibrating Camera Offsets to Facilitate Object Position Determination Using Triangulation” issued on July 19, 2005.

Following the camera offset angle calibration routine, a surface detection routine is performed to enhance determination as to whether a pointer is in contact with the touch surface 60 at a given point or hovering above the touch surface.

With rectangular coordinates of a pointer in the plane of the touch surface 60 accurately known from the camera offset angle calibration, the orientation of the touch surface 60 as seen by each digital camera 63 can be determined. This is necessary due to the fact that the digital cameras do not just see along the plane of the touch surface 60 but also in a direction perpendicular to it. To some degree, each digital camera 63 looks downward into the touch surface 60. Figure 14 generally shows the shape of the touch surface 60 as seen by a digital camera 63. Because of this, it is desired to define a "vertical" coordinate z which describes the touch surface location as a function of rectangular coordinates x and y.

The z coordinate of the pointer can be measured from a digital camera image, and hence, z coordinates for pointer positions on the touch surface 60 can be determined. This vertical calibration becomes a matter of fitting the z coordinate data for given rectangular coordinates x and y. The vertical calibration can be described as a surface of the form:

$$z(x,y) = Ax + By + Cx^2 + Dy^2 + Exy + F \tag{0.1}$$

Note that if the coefficients C, D, and E are zero, this becomes a plane. The fit is easily computed as equation (0.1) represents a linear least-squares problem. The corresponding matrix takes the form:

$$\begin{bmatrix} x_1 & y_1 & x_1^2 & y_1^2 & x_1y_1 & 1 \\ x_2 & y_2 & x_2^2 & y_2^2 & x_2y_2 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_N & y_N & x_N^2 & y_N^2 & x_Ny_N & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \\ D \\ E \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix}$$

In order to fit the rectangular coordinates x and y to the equation (0.1) to determine the coefficients A to E, the Moore-Penrose pseudo-inverse method that is based on singular value decomposition (SVD) is used to determine a minimum-norm least squares solution.

As will be appreciated, a matrix can always be decomposed in the following way:

$$A = USV^T \tag{0.2}$$

Matrix A can have any shape. The matrices U and V are orthogonal matrices, meaning that:

$$U^T U = I = V^T V$$

The diagonal matrix S is composed entirely of the singular values of matrix A, which are related to the squares of the eigenvalues of matrix A. The importance of the singular value decomposition (SVD) lies in the fact that with it, the inverse of matrix A can always be computed. Moreover, it is possible to control this inversion when a poorly determined problem is encountered. Consider the system of linear equations:

$$A\vec{x} = \vec{b}$$

whose solution would be:

$$\vec{x} = A^{-1}\vec{b}$$

SVD allows the inverse of matrix A to be written as:

$$A^{-1} = VS^{-1}U^T \tag{0.3}$$

since both matrices U and V are orthogonal. In a poorly determined situation, some of the singular values will be very small, so that when matrix S^{-1} is formed, large values will be produced, which is not desirable. In this case, the inverses of the smallest singular values are set to zero. This has the effect of eliminating the poorly determined part of the solution. For least-squares problems, this is a powerful tool. The usual normal equations method for least-squares problems is based on solving:

$$A^T A \vec{x} = A^T \vec{b} \tag{0.4}$$

$$\vec{x} = \left(A^T A\right)^{-1} A^T \vec{b}$$

in the over-determined case, and solving:

$$\vec{x} = A^T \left(AA^T\right)^{-1} \vec{b} \tag{0.5}$$

in the under-determined case. As will be appreciated, during fitting of the system of equations to equation (0.1), the same method is used as is used during determination of the camera offset angles δ. Since the same procedure is used, memory usage and processing speed is maintained at desired levels.

With the coefficients A through E known, the z coordinate for any given (x,y) point on the touch surface can be calculated and thus, a determination can be made as to whether a pointer is contacting the touch surface 60 or hovering above it.

With the touch system 50 calibrated, during operation each digital camera 63 acquires images of the touch surface 60 within its field of view. The images are acquired by the image and lens assembly 80 at intervals in response to the clock signals received from the DSP 84. Each image acquired by the image and lens assembly 80 is sent to the FIFO buffer 82. The DSP 84 in turn reads each image from the FIFO buffer 82 and processes the image. To avoid processing significant numbers of pixels containing no useful information, only a subset of the pixels in the acquired image are processed as is shown in Figure 9.

During processing of an image acquired by a digital camera 63, the DSP 84 executes a processFrame routine as shown in Figure 6. When an image is available for processing (step 120), a check is made to determine if the image has been captured for the purpose of adjusting the digital camera 63 (step 122). If the image has been acquired for the purpose of exposure adjustment, an exposureControl routine is called (step 124) to adjust the exposure of the digital camera 63. Following this, the DSP 84 awaits receipt of the next image available for processing.

At step 122, if the image has not been captured for the purpose of adjusting the exposure of the digital camera 63, a check is made to determine if the image has been captured for the purpose of replacing a background image (step 126). If the image has been acquired for the purpose of background image replacement, a captureBackground routine is called (step 128) and the acquired image is used as the background image. This is done if a digital camera acquires an image and sends a PIP to the master controller indicating that a pointer is in the image when it is actually noise. Replacing the background image effectively inhibits the digital camera from falsely identifying a pointer in future PIPs. Following this, the DSP 84 awaits receipt of the next image available for processing.

At step 126, if the image has not been captured for the purpose of background image replacement, a copyICur routine is called by the DSP 84 (step 130). During this routine, the current acquired image is copied into memory and is used to update the background image as well as to form a difference image representing the differences between the current acquired image and the background image.

After completion of the copyICur routine, a segmentPointer routine is called (step 132) to determine if a pointer is in the acquired image and if so to determine the location of the pointer relative to the touch surface 60 and whether the pointer is in contact with the touch surface 60 or hovering above it. The segmentPointer routine 132 also allows changing light conditions to be detected. Following the segmentPointer routing 132, the DSP 84 calls a fillPIP routine (step 134) to place the pointer and light condition information into a PIP for transmission to the master controller 54. Thereafter, the DSP 84 awaits receipt of the next image available for processing.

Figure 7 illustrates the steps performed by the DSP 84 during execution of the segmentPointer routine 132. As can be seen, when the DSP 84 executes the segmentPointer routine, the DSP 84 calls a findPointer routine to determine if a pointer is in the acquired image and if so, the position of the pointer in the current acquired image (step 140). Upon completion of the findPointer routine 140, the DSP 84 calls an updateBackground routine to update the background image thereby to deal with changes in lighting conditions (step 142).

During execution of the updateBackground routine, the DSP 84 continuously updates the background image using the equation:

$$B_{n+1}(i,j) = (l-a)\,B_n(i,j) + aI(i,j) \qquad (0.6)$$

where:

B_{n+1} is the new background image;

B_n is the current background image;

I is the current acquired image;

i,j are the row and column coordinates of the background image pixels being updated; and

a is a number between 0 and l that indicates the degree of learning that should be taken from the current acquired image I. The larger the value of a, the faster the background image is updated.

After the updateBackground routine 142 has been executed, the intensity difference between the current acquired image and the background image is calculated by the DSP 84. This information is sent to the master controller 54 to enable the master controller to determine if the digital camera 63 needs to be re-exposed. This would be required if a drastic change in lighting conditions occurred (i.e. environment lighting was switched on or off). When re-exposure of the digital camera 63 is required, the master controller 54 sends a command PIP to the digital camera 63 instructing the digital camera to acquire an image for exposure adjustment.

Figure 8 illustrates the steps performed by the DSP 84 during execution of the findPointer routine 140. As can be seen, when the DSP 84 executes the findPointer routine 140, the DSP 84 clears pointer location and pointer tip parameters x and z respectfully (step 150). Thereafter a vertical intensity histogram is built (step 152). During this stage, the difference image representing differences between the current image and background image is formed and pixel intensities in the difference image are summed by column. In this manner a 640 x 1 vector is formed that represents the sum of each column in the 640 x 20 difference image. Thus, the first element in the 640 x 1 vector represents the sum of the 20 pixels in the first column of the 640 x 20 difference image, the second element in the 640 x 1 vector represents the sum of the 20 pixel in the second column of the 640 x 20 difference image and so on. Further specifics of this process can be found in the article entitled " A smart camera application: DSP – based people detection and tracking" authored by V. Cheng et al and published in the SPIE Journal of Electronic Imaging July, 2000.

Following the creation of the vertical intensity histogram at step 152, the pointer location parameter x is determined by finding the column in the vertical intensity histogram with the highest intensity above a noise threshold (step 154). The column is used as the center of a region of interest (ROI) to be processed with the width of the ROI being equal to the base of the peak formed by the vertical intensity histogram (see Figure 10). If no column has an intensity above the noise threshold, it is assumed no pointer is within the acquired image.

When a pointer location parameter x is determined, the DSP 84 analyses the ROI to determine the pixel row where the pointer tip is located and determine whether that row represents a touch surface contact or hover (step 156). Specifically, the DSP 84 creates a binary mask in the ROI so that white pixels represent the pointer and black pixels represent the background as shown in Figure 12. From the mask, the medium line of the pointer and the pointer tip location z can be easily calculated.

During the fillPIP routine 134, the DSP 84 uses the pointer and light condition information acquired during execution of the segmentPointer routine 132 and creates a PIP to reduce the acquired image to a small set of data thereby to provide bandwidth economy. The PIP is in the form of a six (6) word packet, with each word in the packet being sixteen (16) bits. The PIP typically takes the form:

Header	Data	Checksum

The header portion of the PIP is typically sixteen (16) bits and includes a determination/source field, a data type field, an image frame number field, a sequence number field and a packet number field. The destination/source field identifies the PIP destination and the PIP source. If the PIP is generated by the master controller 54, the destination may be a single digital camera 63 or all digital cameras. The data type indicates whether the PIP relates to pointer information or other information such as status and diagnostic information. The image frame number field stores a number so that images from each digital camera 63 are processed by the master controller 54 in sequence. The sequence number field stores a number that relates the PIP to other PIPs. The packet number field stores a number identifying the packet.

The data portion of the PIP is typically sixty-four (64) bits and includes a pointer ID field, a pointer location parameter field, a pointer tip parameter field, a contact state field and a goodness of pointer field. The pointer ID field stores an identifier for the pointer to allow multiple pointers to be tracked. The pointer location parameter field stores the *x*-value calculated by the DSP 84. The pointer tip parameter field stores the z-value calculated by the DSP 84. The contact state field stores a value that indicates whether the pointer is in contact, out of contact or possibly in contact with the touch surface 60. The goodness of pointer field stores a statistical value on the likelihood that a detected pointer is real.

The checksum portion of the PIP is used to ensure PIP transmission integrity. If PIP checksum errors are infrequent, the PIPs exhibiting checksum errors are ignored by the destination device.

Status PIPs that do not relate to pointer information have a different form then the above-identified described PIPs. For PIPs of this nature, the data portion includes an instruction type field, an instruction code field and a data field. The instruction type field identifies whether the instruction type is an instruction to be performed or a status request. The instruction code field stores the actual instruction or status request identifier. The data field stores data that varies depending on the type of instruction. Examples of status PIPs include frame header PIPs, command PIPs and error message PIPs.

A frame header PIP typically includes the number of pointer PIPs that are to follow for a current acquired image with statistics for the current image such as intensity variance between the current acquired image and a previous image. A command PIP issued by the master controller 54 may instruct a digital camera to adjust one or more of its settings such as exposure or capture an image to be used as a new background image. An error PIP may pass an error condition from a digital camera 63 to the master controller 54 for storage in an error log.

Each digital camera 63 processes each image it acquires in the manner described above in response to each clock signal generated by its DSP 84. The PIPs created by the DSPs 84 are only sent to the master controller 54 when the digital cameras 63 are polled by the master controller 54.

When the master controller 54 polls the digital cameras 63, frame sync pulses are sent to the digital cameras 63 to initiate transmission of the PIPs created by the DSPs 84. Upon receipt of a frame sync pulse, each DSP 84 transmits the PIP to the master controller 54 over the data bus. The PIPs transmitted to the master controller 54 are received via the serial port 96 and auto-buffered into the DSP 90.

After the DSP 90 has polled the digital cameras 63 and has received PIPs from each of the digital cameras 63 that include pointer information, the DSP 90 processes the PIPs using triangulation to determine the location of the pointer relative to the touch surface 60 in (x,y) coordinates. Specifically, the PIPs from pairs of digital cameras 63 are processed using triangulation.

Figure 11 shows that two angles ϕ_{cam_1} and ϕ_{cam_2} are needed to triangulate the position (x_0, y_0) of a pointer relative to the touch screen 60. The PIPs generated by each digital camera 63 include a number θ (see Figure 12) identifying the median line or tip of the pointer. When the master controller 54 receives a PIP from a digital camera 63, the master controller uses the number representing the median line or tip of the pointer and the field of view of the digital camera to calculate an angle ϕ_{cam} using the equation:

$$\tan\phi_{cam} = \frac{2\left(\frac{x}{a}\right)\tan\frac{FOV}{2}}{1-\left(2\frac{x}{a}-1\right)\tan^2\frac{FOV}{2}} \qquad (0.7)$$

where:

x is the number representing the median line or tip of the pointer; and

a is the total length enclosed by the field of view (FOV) of the digital camera at a distance from the camera.

The calculated angle ϕ_{cam} is equal to the angle formed between the extremity of the field of view extending beyond the designated peripheral edge of the touch surface 60 of the digital camera 63 that generated the PIP and a line extending from the optical axis of the digital camera that intersects the pointer within the acquired image. Preferably, the extremity of the field of view extends beyond the designated peripheral edge (i.e. in this case the x-axis) of the touch surface 60 within the field of view by a known amount. However, in almost all cases the angular offset δ_{cam} scan of each digital camera 63 is different and unknown.

Once the master controller 54 calculates the angle ϕ_{cam}, the master controller 54 uses the camera offset angle δ_{cam} determined during the camera offset calibration to adjust the angle ϕ_{cam}. With the two angles available and with the angles ϕ_{cam} adjusted, the master controller 54 uses the angles ϕ_{cam} to determine the position of the pointer relative to the touch surface 60 using triangulation.

In this embodiment, since the touch screen 52 includes four digital cameras 63, six pairs of digital cameras can be used for triangulation. The following discussion describes how a pointer position is determined by triangulation for each pair of the digital cameras 63.

In order to determine a pointer position using the PIPs received from the digital cameras 63 along the left side of the touch screen 52, the following equations are used to determine the (x_0, y_0) coordinates of the pointer position given the angles ϕ_0 and ϕ_1 for the upper and lower digital cameras:

$$x_0 = \frac{h}{w} \times \frac{1}{\tan(\phi_0) + \tan(\phi_1)} \tag{0.8}$$

$$y_0 = \frac{\tan(\phi_0)}{\tan(\phi_0) + \tan(\phi_1)} \tag{0.9}$$

where:

h is the height of the touch screen 52 i.e. the vertical distance from digital camera focal point-to-focal point;

w is the width of the touch screen 52 i.e. the horizontal distance from digital camera focal point-to-focal point; and

ϕ_i is the angle with respect to the horizontal, measured using digital camera i and equation (0.7).

For the digital cameras 63 along on the right side of the touch screen 52, the following equations are used to determine the (x_0, y_0) coordinates of the pointer position given the angles ϕ_2 and ϕ_3 for the upper and lower digital cameras:

$$x_0 = 1 - \frac{h}{w} \times \frac{1}{\tan(\phi_2) + \tan(\phi_3)} \tag{0.10}$$

$$y_0 = 1 - \frac{\tan(\phi_2)}{\tan(\phi_2) + \tan(\phi_3)} \tag{0.11}$$

The similarity between equations (0.8) and (0.10), i.e. equation (0.10) = 1 - equation (0.8) once angles ϕ_2 and ϕ_3 have been substituted into equation (0.8) for angles ϕ_1 and ϕ_2 respectively should be apparent. Equations (0.9) and (0.11) are related in a similar manner.

In order to determine a pointer position using the digital cameras 63 along the bottom of the touch screen 52, the following equations are used to determine the (x_0, y_0) coordinates of the pointer position given the angles ϕ_0 and ϕ_3 for bottom left and bottom right digital cameras:

$$x_0 = \frac{\tan(\phi_3)}{\tan(\phi_0) + \tan(\phi_3)} \qquad (0.12)$$

$$\begin{aligned} y_0 &= \frac{w}{h} \times \frac{\tan(\phi_3)}{\tan(\phi_0) + \tan(\phi_3)} \times \tan(\phi_0) \\ &= \frac{w}{h} \times x_0 \times \tan(\phi_0) \end{aligned} \qquad (0.13)$$

In order to determine a pointer position using the digital cameras 63 along the top of the touch screen 52, the following equations are used to determine the (x_0, y_0) coordinates of the pointer position given the angles ϕ_1 and ϕ_2 for the top left and top right digital cameras:

$$x_0 = \frac{\tan(\phi_2)}{\tan(\phi_1) + \tan(\phi_2)} \qquad (0.14)$$

$$\begin{aligned} y_0 &= 1 - \frac{w}{h} \times \frac{\tan(\phi_2)}{\tan(\phi_1) + \tan(\phi_2)} \times \tan(\phi_1) \\ &= 1 - \frac{w}{h} \times x_0 \times \tan(\phi_1) \end{aligned} \qquad (0.15)$$

The similarity between equations (0.12) and (0.14), i.e. equation (0.14) = equation (0.12) once angles ϕ_1 and ϕ_2 have been substituted into equation (0.12) for angles ϕ_0 and ϕ_3 should be apparent. Equations (0.13) and (0.15) have the following relationship: equation (0.15) = 1 - equation (0.13) once angles ϕ_1 and ϕ_2 have been substituted into equation (0.13) for angles ϕ_0 and ϕ_3 respectively.

In order to determine a pointer position using the digital cameras 63 across the bottom left to top right corner diagonal, the following equations are used to determine the (x_0, y_0) coordinates of the pointer position given the angles ϕ_0 and ϕ_2 for bottom left and top right digital cameras:

$$x_0 = \frac{\frac{h}{w} - \tan(\phi_2)}{\tan(\phi_0) - \tan(\phi_2)} \tag{0.16}$$

$$y_0 = \frac{1 - \frac{w}{h} - \tan(\phi_2)}{\tan(\phi_0) - \tan(\phi_2)} \times \tan(\phi_0) \tag{0.17}$$

In order to determine a pointer position using the digital cameras 63 across the bottom right to top left diagonal, the following equations are used to determine the (x_0, y_0) coordinates of the pointer position given the angles ϕ_1 and ϕ_3 for the bottom right and top left digital cameras:

$$x_0 = \frac{\frac{h}{w} - \tan(\phi_3)}{\tan(\phi_1) - \tan(\phi_3)} \tag{0.18}$$

$$y_0 = 1 - \frac{1 - \frac{w}{h} - \tan(\phi_3)}{\tan(\phi_1) - \tan(\phi_3)} \times \tan(\phi_1) \tag{0.19}$$

The similarity between equations (0.16) and (0.18), i.e. equation (0.18) = equation (0.16) once angles ϕ_1 and ϕ_3 have been substituted into equation (0.16) for angles ϕ_0 and ϕ_2 should be apparent. Equations (0.17) and (0.19) have the following relationship: equation (0.19) = 1 - equation (0.17) once angles ϕ_1 and ϕ_3 have been substituted into equation (0.17) for angles ϕ_0 and ϕ_2 respectively.

As will be appreciated, the above equations generate the coordinates x_0 and y_0 on a scale of [0, 1]. Therefore, any appropriate coordinate scale can be reported by multiplying x_0 and y_0 by the maximum X and maximum Y values respectively.

In the present embodiment, the DSP 90 calculates the pointer position using triangulation for each digital camera pair excluding the diagonal pairs. The resulting pointer positions are then averaged and the resulting pointer position coordinates are queued for transmission to the personal computer 56 via the serial port 98 and the serial line driver 94.

With the (x,y) position of a pointer known by triangulation, using the coefficients A to E calculated during the surface detection calibration, the z coordinate corresponding to the (x,y) position can be determined using equation (0.1). Calculating the z coordinate and comparing the z coordinate with the z parameter in the PIP provides an indication as to whether the pointer is hovering above the touch surface 60 or is in actual contact with the touch surface.

If desired, pointer velocity v and angle can be calculated by the DSP 90 as shown in Figure 13. The velocity of the pointer is calculated by examining the changes in the z-position (or x-intercept) of the pointer in successive PIPs and knowing the camera frame rate. For example, if the camera frame rate is 200 frames per second and the z-position changes by 1 pixel row per frame, the pointer velocity is 200 pixels per second.

The angle of the pointer can be determined due to the fact that the PIP includes the x-intercept at pixel rows 0 and 19 of the median line. Since the x distance (the difference between x-intercepts) and the y distance (the number of pixel rows) are known, all of the information necessary to calculate the pointer angle is available.

If desired, a Kalman filter (essentially a recursive least-squares method) may be used to effectively "track" the pointer when it is within a certain distance of the touch surface 60. To do this, it is necessary to define a system equations or model to be used in the filter. Since the master controller 54 is able to provide both the position z and velocity v of the pointer, the following description can be used:

$$z = z_0 + vt$$
$$v = v$$

The second of these equations is required as the filter has to know what to do with the velocity, and also since both z and v are measurable. Define the state vector as:

$$[z \;\; v]^{\mathrm{T}}$$

To relate the state of the system at two successive times n and n+1, write the system equations as a matrix difference equation:

$$\begin{bmatrix} z \\ v \end{bmatrix}_{n+1} = \begin{bmatrix} 1 & dt \\ 0 & 1 \end{bmatrix}\begin{bmatrix} z \\ v \end{bmatrix}_n + \begin{bmatrix} \sigma_z \\ \sigma_v \end{bmatrix}$$

or in matrix notation,

$$\hat{x}_{n+1} = A\hat{x}_n + \sigma$$

Here, dt denotes the time interval between successive time steps. Also introduced here on the RHS is the "process noise" term. It is purely formal, but part of the Kalman filter method. It is also necessary to specify how a measurement is introduced into the procedure. This is done via the matrix equation:

$$z_n = Hx_n + w$$

where z_n is a measurement of position and velocity, H is a "measurement matrix" which is taken to be an identity matrix, x_n is the state vector and w is measurement noise. Essentially, it is assumed that the measurements are noisy versions of the state vector. It is also necessary to define a covariance matrix associated with w. If the measurement error in z is 0.5 pixel, then the covariance matrix is:

$$R = (0.5)^2\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

A similar matrix Q is required for the process noise introduced above, but as it is somewhat arbitrary, it may be treated as a tuning parameter for the filter. In this example, the matrix Q is taken to be an identity matrix multiplied by a factor of order unity or less. With the above established, there is sufficient information to start the filter process. The first (prediction) step is:

$$\hat{x}_{k+1}(-) = A\hat{x}_k(+)$$
$$P_k(-) = AP_{k-1}(+)A^T + Q_{k-1}$$

Here, the (-) notation implies that a measurement has not yet been made while (+) does (but in this case the (+) refers to the previous step). Also, the matrix equation for matrix P predicts a covariance matrix. The next step is the filter gain computation:

$$K_k = P_k(-)H_k^T\left[H_kP_k(-)H_k^T + R_k\right]^{-1}$$

Once a measurement is made, the state estimate and its covariance can be updated:

$$\hat{x}_k(+) = \hat{x}_k(-) + K_k\left[z_k - H_k x_k(-)\right]$$
$$P_k(+) = \left[P_k^{-1}(-) + H_k^T R_k^{-1} H_k\right]^{-1}$$

It is this estimate of the state x that is used to determine whether or not contact with the touch surface has occurred. Note here that the matrices H and R are both constant with time, and that only matrices K and P change (in fact, P approaches a constant matrix). An additional simplification occurs in that there is no control process involved.

The results of a Matlab simulation of a Kalman filter using a set of measurements representing a pointer approaching the touch surface 60 at a constant velocity was performed. Figures 15 and 16 illustrate the simulation, with a time step dt of 0.1 sec and a measurement precision of 0.5 pixel. The open symbols represent the data, and the lines the state estimate from the Kalman filter. Clearly, the state estimate follows the data quite well.

A second Matlab simulation was performed to take into account both vertical (z) and horizontal (x) motion of a pointer. This simulation is basically two similar Kalman filters operating together in a "parallel" fashion. The formulation is exactly the same, except twice the number of variables need to be considered. Figures 17a to 17d show the results of the simulation and represent movement of a pointer towards the touch surface 60 at constant velocity and at a slowly-varying x position (i.e. the person's hand is unsteady).

Although the touch system 50 has been described as including a projector to present images on the touch screen, those of skill in the art will appreciate that this is not required. The touch screen 52 may be transparent or translucent and placed over a display unit so that the display presented on the display unit is visible through the touch screen. Also, the touch screen need not be a rectangular sheet of material bordered by a frame. The touch screen may in fact be virtually any surface within overlapping fields of view of two or more digital cameras.

Also, although the touch system 50 is described as including a master controller separate from the digital cameras, if desired one of the digital cameras can be conditioned to function as both a camera and the master controller and poll the other digital cameras for PIPs. In this case, it is preferred that the digital camera functioning as the master controller includes a faster DSP 84 than the remaining digital cameras.

In addition, although the surface detection routine is described as determining the coefficients A to E to be used with equation (0.1) to calculate the z coordinates of the pointer at a given point (x,y) relative to the touch screen, during the surface detection routine, the master controller 54 can be programmed to calculate a z coordinate for unique (x,y) regions of the touch surface and store the z coordinates in a look-up table (LUT). In this instance, when a pointer appears in images captured by the digital cameras and the (x,y) position of the pointer relative to the touch surface is determined, a decision can be made as to whether the pointer is in contact with the touch surface by comparing the z coordinate in the LUT corresponding with the (x,y) region in which the pointer is located, with the pixel row of the image sensor and lens assembly at which the pointer tip is located.

As described above, the master controller 54 calculates or looks up the z coordinates of the touch surface for each digital camera and compares the z coordinates with the pointer tip location z to determine if the pointer is in actual contact with the touch surface. However, those of skill in the art will appreciate that the DSPs 84 in the digital cameras may include image processing software to determine if the pointer is in actual contact with the touch surface. This image processing can be preformed in conjunction with or instead of the master controller pointer contact determination.

Although a preferred embodiment of the present invention has been described, those of skill in the art will appreciate that variations and modifications may be made without departing from the spirit and scope thereof as defined by the appended claims.

What is claimed is:

1. A passive touch system comprising:

a touch surface;

at least two cameras associated with said touch surface, said at least two cameras acquiring images of said touch surface from different vantages and having overlapping fields of view;

a digital signal processor associated with each camera, the digital signal processors associated with said at least two cameras selecting pixel subsets of images acquired by said at least two cameras and processing pixel data acquired by the selected pixel subsets to generate pointer characteristic data when a pointer exists in said acquired images; and

a master digital signal processor in communication with said digital signal processors, said master digital signal processor receiving pointer characteristic data from said digital signal processors and triangulating the pointer characteristic data to determine the location of said pointer relative to said touch surface.

2. A passive touch system according to claim 1 wherein said digital signal processors generate pixel characteristic data when the pointer is in contact with said touch surface and when said pointer is hovering above said touch surface.

3. A passive touch system according to claim 1 or 2 wherein said selected pixel subsets are determined during an alignment routine.

4. A passive touch system according to any one of claims 1 to 3 wherein said pointer characteristic data includes a median line of the pointer.

5. A passive touch system according to claim 4 wherein said pointer characteristic data is packaged by said digital signal processors into pointer information packets (PIPs).

6. A passive touch system according to any one of claims 1 to 5 wherein

said touch surface is substantially rectangular and wherein a camera is located adjacent each corner of said touch surface.

7. A passive touch system according to any one of claims 1 to 6, wherein each said camera comprises a CMOS digital camera having a selectable pixel array.

8. A passive touch system according to claim 7, wherein said selected pixel subsets include contiguous rows of pixels.

9. A passive touch system according to claim 7, wherein the selected pixel subset for each digital camera is aligned so that the selected pixel subset looks substantially along the plane of said touch surface.

10. A passive touch system according to claim 9, wherein the row of pixels in the selected pixel subset for each digital camera, that acquires pixel data corresponding to a tip of said pointer on said touch surface, is determined.

11. A passive touch system according to any one of claims 6 to 10, wherein during said processing, said master digital signal processor triangulates pointer characteristic data associated with multiple pairs of digital cameras.

12. A passive touch system according to claim 11, wherein said master digital signal processor averages the triangulated pointer characteristic data to determine the location of said pointer relative to said touch surface.

13. A passive touch system according to any one of claims 1 to 12, wherein communication between said master digital signal processor and said digital signal processors is bidirectional.

14. A passive touch system according to claim 13, wherein said digital signal processors also communicate diagnostic data to said master digital signal processor.

15. A passive touch system according to any one of claims 1 to 14 further comprising a computer coupled to said master digital signal processor, said computer receiving pointer location data from said master digital signal processor and processing the same.

16. A touch system comprising:

at least two CMOS digital cameras associated with a touch surface, said at least two digital cameras acquiring images of said touch surface from different vantages and having overlapping fields of view; and

processing structure receiving and processing image data acquired by said at least two digital cameras to detect the existence of a pointer in said images and to determine the location of said pointer relative to said touch surface, said processing structure processing the image data of each digital camera by compensating for an angle between (i) a field of view of said each digital camera and (ii) a periphery of said touch surface as a reference.

17. A touch system according to claim 16, wherein each of said at least two digital cameras has a selectable pixel array.

18. A touch system according to claim 17, wherein a subset of pixels in the selectable pixel array of each digital camera provides pixel data to said processing structure.

19. A touch system according to claim 18, wherein the pixel subset of each digital camera includes contiguous rows of pixels.

20. A touch system according to claim 18 or 19, wherein the pixel subset of each digital camera is aligned so that the pixel subset of each digital camera looks substantially along the plane of said touch surface.

21. A touch system according to claim 20, wherein the row of pixels in the

pixel subset of each digital camera that acquires pixel data corresponding to a tip of said pointer on said touch surface is determined to enable said processing structure to determine pointer contact with said touch surface and pointer hover over said touch surface.

22. A touch system according to any one of claims 16 to 21, wherein said touch surface is substantially rectangular wherein a digital camera is positioned adjacent each corner of said touch surface.

23. A touch system according to claim 22, wherein said processing structure triangulates image data from multiple pairs of digital cameras to determine the location of the pointer relative to said touch surface.

24. A touch system according to any one of claims 16 to 23, wherein said processing structure includes a plurality of processing stages, said processing stages processing image data in a manner to stage bandwidth.

25. A touch system according to claim 24, wherein one of said processing stages includes a plurality of digital signal processors each associated with a respective one of said digital cameras, said digital signal processors processing pixel data from pixel subsets of said digital cameras and generating pointer parameter data.

26. A touch system according to claim 25, wherein a second of said processing stages includes a master digital signal processor receiving said pointer parameter data from said digital signal processors, said master digital signal processor triangulating said pointer parameter data.

27. A touch system according to claim 26, wherein a third of said processing stages includes a computer receiving the location of said pointer relative to said touch surface from said master digital signal processor.

28. A touch system according to claim 26 or 27, wherein communication

between said master digital signal processor and said digital signal processors is bi-directional.

29. A touch system according to any one of claims 26 to 28, wherein said digital signal processors also communicate digital camera diagnostic data to said master digital signal processor.

30. A touch system according to any one of claims 16 to 29, wherein said digital cameras are arranged relative to said touch surface so that the fields of view thereof extend beyond at least one peripheral edge of said touch surface.

31. A touch system comprising:

at least two camera devices associated with a touch surface, said at least two camera devices acquiring images of said touch surface from different vantages and having overlapping fields of view; and

processing structure receiving and processing image data acquired by said at least two camera devices to detect the existence of a pointer in said images and to determine the location of said pointer relative to said touch surface, wherein said processing structure includes first and second processing stages, said first processing stage processing pixel data from said at least two camera devices, said second processing stage processing image data from said first processing stage to determine the location of the pointer, wherein said first processing stage includes a plurality of digital signal processors each associated with a respective one of said camera devices, said digital signal processors processing pixel data from pixel subsets of said camera devices and generating pointer parameter data, wherein said touch surface is substantially rectangular and wherein camera devices are positioned adjacent each corner of said touch surface, each camera device being arranged relative to said touch surface so that the field of view thereof extends beyond at least one peripheral edge of said touch surface.

32. A touch system according to claim 31 wherein, said second processing stage includes a master digital signal processor receiving said pointer parameter data

from said digital signal processors, said master digital signal processor triangulating said pointer parameter data.

33. A touch system according to claim 32, wherein said processor further includes a third processing stage comprising a computer receiving the location of said pointer relative to said touch surface from said master digital signal processor.

34. A touch system according to claim 32 or 33, wherein communication between said master digital signal processor and said digital signal processors is bi-directional.

35. A touch system according to any one of claims 32 to 34, wherein said digital signal processors also communicate camera device diagnostic data to said master digital signal processor.

36. A touch system according to any one of claims 32 to 35, wherein said master digital signal processor further calculates the velocity of said pointer as said pointer is brought towards said touch surface.

37. A touch system according to claim 36, wherein said velocity is calculated by determining the product of the change in pointer tip position within said pixel subset in successive images acquired by said digital cameras and the frame rates of the digital cameras.

38. A touch system according to any one of claims 32 to 37, wherein said master digital signal processor further calculates the angle of said pointer as said pointer is brought towards said touch surface.

39. A touch system according to claim 38, wherein said angle is determined using the position of said pointer in extreme rows of pixels in said pixel subsets and the number of pixel rows in said pixel subsets.

40. A touch system according to any one of claims 31 to 39, wherein each said camera device comprises a CMOS digital camera having a selectable pixel array.

41. A touch system comprising:

at least two CMOS digital cameras associated with a touch surface and having overlapping fields of view, each of said at least two CMOS digital cameras having an array of pixels, said at least two CMOS digital cameras acquiring images of said touch surface from different vantages substantially simultaneously, said digital cameras being arranged so that the field of view of each digital camera extends beyond at least one peripheral edge of said touch surafce; and

processing structure receiving and processing pixel data acquired by a window of contiguous pixels within the pixel array of said at least two CMOS digital cameras to detect the existence of a pointer in said images and to triangulate the location of said pointer relative to said touch surface.

42. A touch system according to claim 41, wherein the window of contiguous pixels used to provide said pixel data from each digital camera is adjustable along said pixel array.

43. A touch system according to claim 42, wherein the window of pixels of each digital camera is aligned so that pixel data corresponding to a tip of said pointer on said touch surface is acquired.

44. A touch system according to claim 43, wherein the row of pixels in the window of each digital camera, that acquires the pixel data corresponding to said pointer tip on said touch surface, is determined to enable said processing structure to determine pointer contact with said touch surface and pointer hover over said touch surface.

45. A touch system according to any one of claims 41 to 44, further comprising a frame surrounding said touch surface, said frame supporting a digital camera adjacent each corner thereof.

46. A touch system according to claim 45, wherein said processing structure triangulates pixel data from multiple pairs of digital cameras to determine the location of the pointer relative to said touch surface.

47. A touch system according to any one of claims 41 to 46, wherein said processing structure includes a plurality of processing stages, said processing stages processing image data in a manner to stage bandwidth.

48. A touch system according to claim 47, wherein one of said processing stages includes a plurality of digital signal processors each associated with a respective one of said digital cameras, said digital signal processors processing pixel data from pixel subsets of said digital cameras and generating pointer parameter data.

49. A touch system according to claim 48, wherein a second of said processing stages includes a master digital signal processor receiving said pointer data from said digital signal processors, said master digital signal processor triangulating said pointer data.

50. A touch system according to claim 49, wherein a third of said processing stages includes a computer receiving the location of said pointer relative to said touch surface from said master digital signal processor.

51. A camera based position tracking system comprising:

a frame;

a region of interest at least partially within said frame;

at least two cameras associated with said frame, said at least two cameras acquiring images of said region of interest from different vantages and having overlapping fields of view, the field of view of each digital camera extending beyond at least one peripheral edge of said frame;

a digital signal processor associated with each camera, the digital signal processors processing pixel data of pixel subsets of images acquired by said at least two cameras to generate pointer characteristic data when a pointer exists in said acquired images; and

a master digital signal processor in communication with said digital signal processors, said master digital signal processor receiving pointer characteristic data from said digital signal processors and triangulating the pointer characteristic data to determine the location of said pointer.

52. A camera based position tracking system according to claim 51 wherein said digital signal processors generate pixel characteristic data when the pointer is within said region of interest.

53. A camera based position tracking system according to claim 51 or 52 wherein said selected pixel subsets are determined during an alignment routine.

54. A camera based position tracking system according to any one of claims 51 to 53 wherein said pointer characteristic data includes a median line of the pointer.

55. A camera based position tracking system according to claim 54 wherein said pointer characteristic data is packaged by said digital signal processors into pointer information packets (PIPs).

56. A camera based position tracking system according to claim 55 wherein said master digital signal processor polls said digital signal processors for said pointer information packets at a set frequency.

57. A camera based position tracking system according to claim 56 wherein said master digital signal processor polls said digital signal processors for said pointer information packets at a frequency of 70 times per second.

58. A camera based position tracking system according to any one of claims 51 to 57 wherein said frame is substantially rectangular and wherein a camera

is located adjacent each corner of said frame.

59. A camera based position tracking system according to any one of claims 51 to 58, wherein each camera comprises a CMOS digital camera having a selectable pixel array.

60. A camera based position tracking system according to claim 59, wherein said selected pixel subsets include contiguous rows of pixels.

61. A camera based position tracking system according to claim 59, wherein the selected pixel subset for each digital camera is aligned relative to the plane of said frame.

62. A camera based position tracking system according to claim 61, wherein the row of pixels in the selected pixel subset for each digital camera, that acquires pixel data corresponding to a tip of said pointer in the plane of said frame, is determined.

63. A camera based position tracking system according to any one claims 51 to 62, wherein said master digital signal processor triangulates pointer characteristic data associated with multiple pairs of digital cameras.

64. A camera based position tracking system according to claim 63, wherein said master digital signal processor averages the triangulated pointer characteristic data to determine the location of said pointer.

65. A camera based position tracking system according to any one of claims 51 to 64, wherein communication between said master digital signal processor and said digital signal processors is bi-directional.

66. A camera based position tracking system according to claims 65, wherein said digital signal processors also communicate diagnostic data to said master

digital signal processor.

67. A position sensitive interactive system comprising:

at least two CMOS digital camera sensors associated with a region of interest, said at least two digital camera sensors acquiring images of said region of interest from different locations and having overlapping fields of view; and

processing structure receiving subsets of image data acquired by said at least two digital camera sensors prior to processing and only processing data of said subsets to detect the existence of a pointer in said images and to determine the location of said pointer within said region of interest.

68. A position sensitive interactive system according to claim 67, wherein each of said at least two digital camera sensors has a selectable pixel array.

69. A position sensitive interactive system according to claim 68, wherein a subset of pixels in the selectable pixel array of each digital camera sensor provides pixel data to said processing structure.

70. A position sensitive interactive system according to claim 69, wherein the pixel subset of each digital camera sensor includes contiguous rows of pixels.

71. A position sensitive interactive system according to claim 70, wherein a touch surface is positioned in back of and proximate to said region of interest.

72. A position sensitive interactive system according to claim 71, wherein the pixel subset of each digital camera sensor is aligned relative to the plane of said touch surface.

73. A position sensitive interactive system according to claim 72, wherein the row of pixels in the pixel subset of each digital camera sensor that acquires pixel data corresponding to a tip of said pointer within said region of interest is determined to enable said processing structure to determine pointer contact with said touch

surface and pointer hover over said touch surface.

74. A position sensitive interactive system according to any of claims 71 to 73 wherein said touch surface is substantially rectangular.

75. A position sensitive interactive system according to claim 74, wherein a digital camera sensor is positioned adjacent each corner of said touch surface.

76. A position sensitive interactive system according to any one of claims 67 to 70, wherein said region of interest is at least partially bounded by a frame.

77. A position sensitive interactive system according to claim 76, wherein said frame is substantially rectangular.

78. A position sensitive interactive system according to claim 77, wherein a digital camera sensor is positioned adjacent each corner of said frame.

79. A position sensitive interactive system according to claim 78, wherein the pixel subset of each digital camera sensor is aligned relative to the plane of said frame.

80. A position sensitive interactive system according to claim 79, wherein a touch surface is positioned in back of and proximate to said region of interest.

81. A position sensitive interactive system according to claim 80, wherein the pixel subset of each digital camera sensor is aligned relative to the plane of said touch surface.

82. A position sensitive interactive system according to claim 81, wherein the row of pixels in the pixel subset of each digital camera sensor, that acquires pixel data corresponding to a tip of said pointer within said region of interest is determined to enable said processing structure to detect pointer contact with said touch surface

and pointer hover over said touch surface.

83. A position sensitive interactive system according to claim 76, wherein said digital camera sensors are arranged relative to said frame so that the fields of view thereof extend beyond at least one peripheral edge of said frame.

84. A position sensitive interactive system according to claim 83, wherein said processing structure includes a plurality of processing stages.

85. A position sensitive interactive system according to claim 84 wherein one of said processing stages includes a plurality of digital signal processors each associated with a respective one of said digital camera sensors, said digital signal processors processing pixel data from pixel subsets of said digital camera sensors and generating pointer parameter data.

86. A position sensitive interactive system according to claim 85, wherein during said one processing stage creates a difference image.

87. A position sensitive interactive system according to claim 86, wherein said one processing stage filters background features from said difference image.

88. A position sensitive interactive system according to claim 86, wherein said one processing stage filters noise from said difference image.

89. A position sensitive interactive system according to claim 85, wherein a second of said processing stages includes a master digital signal processor receiving said pointer parameter data from said digital signal processors, said master digital signal processor triangulating said pointer parameter data.

90. A position sensitive interactive system according to claim 89, wherein a third of said processing stages includes a computer receiving the location of said pointer relative to said frame from said master digital signal processor.

91. A position sensitive interactive system according to any one of claims 67 to 90, wherein said processing structure processes image data from multiple pairs of digital camera sensors to determine the location of the pointer.

92. A position sensitive interactive system according to claim 89 or 90, wherein communication between said master digital signal processor and said digital signal processors is bi-directional.

93. A position sensitive interactive system according to claim 92, wherein said digital signal processors also communicate digital camera diagnostic data to said master digital signal processor.

94. A camera based interactive display system comprising:

an interactive display;

a region of interest in front of said interactive display;

at least two camera devices associated with a frame, said at least two camera devices acquiring images of said region of interest from different vantages and having overlapping fields of view; and

processing structure receiving a subset of image data acquired by said at least two camera devices prior to processing and processing the image data subset to detect the existence of a pointer in said images and to determine the location of said pointer, wherein said processing structure includes at least first and second processing stages, said first processing stage only processing image data of said subset from said at least two camera devices, said second processing stage processing image data from said first processing stage to determine the location of the pointer, wherein said first processing stage includes a plurality of digital signal processors each associated with a respective one of said camera devices, said digital signal processors processing pixel data from pixel subsets of said camera devices and generating pointer parameter data, wherein said touch surface is substantially rectangular and wherein camera devices are positioned adjacent each corner of said touch surface, each camera device being arranged relative to said touch surface so that the field of view thereof extends beyond at least one peripheral edge of said touch surface.

95. A camera based interactive display system according to claim 94, wherein said first processing stage creates a difference image of said image data subsets.

96. A camera based interactive display system according to claim 94, wherein said first processing stage maps grayscale pixel data of said difference image to a binary difference image.

97. A camera based interactive display system according to claim 96, whereby grayscale pixel data having a value less than or equal to a threshold value is made black.

98. A camera based interactive display system according to claim 97, whereby grayscale pixel data having a value greater than said threshold value is made white.

99. A camera based interactive display system according to any of claims 96 to 98, wherein said first processing stage filters background features from said binary difference image.

100. A camera based interactive display system according to any one of claims 96 to 98, wherein said first processing stage filters noise from said binary difference image.

101. A camera based interactive display system according to any one of claims 94 to 100 wherein, said second processing stage includes a master digital signal processor receiving said pointer parameter data from said digital signal processors, said master digital signal processor triangulating said pointer parameter data.

102. A camera based interactive display system according to claim 101, wherein said processing structure further includes a third processing stage comprising a computer receiving the location of said pointer from said master digital signal processor.

103. A camera based interactive display system according to claim 101 or 102, wherein communication between said master digital signal processor and said digital signal processors is bi-directional.

104. A camera based interactive display system according to any one of claims 101 to 103, wherein said digital signal processors also communicate camera device diagnostic data to said master digital signal processor.

105. A camera based interactive display system according to any one of claims 101 to 104, wherein said master digital signal processor further calculates the velocity of said pointer as said pointer is brought towards said interactive display.

106. A camera based interactive display system according to claim 105, wherein said velocity is calculated by determining the product of the change in pointer tip position within said pixel subset in successive images acquired by said camera devices and the frame rates of the camera devices.

107. A camera based interactive display system according to claim 106, wherein said frame rates of the camera devices are in excess of 200 frames per second.

108. A camera based interactive display system according to any one of claims 101 to 107, wherein said master digital signal processor further calculates the angle of said pointer as said pointer is brought towards said interactive display.

109. A camera based interactive display system according to claim 108, wherein said angle is determined using the position of said pointer in extreme rows of

pixels in said pixel subsets and the number of pixel rows in said pixel subsets.

110. A camera based interactive display system according to any one of claims 93 to 109, wherein said frame is substantially rectangular and wherein camera devices are positioned adjacent each corner of said frame.

111. A camera based interactive display system according to claim 110, wherein each camera device comprises a CMOS digital camera sensor having a selectable pixel array.

112. A camera based interactive display system according to claim 102, wherein said personal computer processes the location of said pointer as a hover event or as an application execution event.

113. A camera based interactive display system according to any one of claims 93 to 112, wherein the location of said pointer within said region of interest is mapped to coordinates corresponding to said interactive display.

114. A camera based position tracking system comprising:

at least two CMOS digital cameras associated with a region of interest and having overlapping fields of view, each of said at least two CMOS digital cameras having an array of pixels, said at least two CMOS digital cameras acquiring images of (i) said region of interest from different locations and (ii) a frame on a periphery of said region of interest; and

processing structure receiving a window of contiguous pixels within and smaller than the pixel array of said at least two CMOS digital cameras and processing pixel data of the pixels in said window to detect the existence of a pointer in said images and to triangulate the location of said pointer, said processing structure processing the image data of each digital camera by compensating for an angle between (i) the field of view of each digital camera and (ii) a periphery of the frame as a reference.

115. A camera based position tracking system according to claim 114, wherein a master processor transmits clock signals to synchronize said at least two CMOS digital cameras.

116. A camera based position tracking system according to claim 114 or 115, wherein the window of contiguous pixels used to provide said pixel data from each digital camera is moveable along said pixel array.

117. A camera based position tracking system according to claim 116, wherein the window of pixels of each digital camera sensor is aligned so that pixel data corresponding to a tip of said pointer at a predetermined location with respect to said at least two CMOS digital cameras is acquired.

118. A camera based position tracking system according to claim 117, wherein said window of pixels of each digital camera is rotated so that pixel data corresponding to said tip of said pointer is acquired.

119. A camera based position tracking system according to claim 117, wherein said window of pixels of each digital camera is selected so that pixel data corresponding to said tip of said pointer is acquired.

120. A camera based position tracking system according to claim 117, wherein the row of pixels in the window of each digital camera, that acquires the pixel data corresponding to said tip of said pointer, is determined to enable said processing structure to determine a pointer hover event and a pointer application execution event.

121. A camera based position tracking system according to claim 120, wherein said frame supports a digital camera adjacent each corner thereof.

122. A camera based position tracking system according to claim 121, wherein said processing structure triangulates pixel data from multiple pairs of digital cameras to determine the location of the pointer.

123. A camera based position tracking system according to any one of claims 114 to 122, wherein said processing structure includes a plurality of processing stages.

124. A camera based position tracking system according to claim 123, wherein one of said processing stages includes a plurality of digital signal processors each associated with a respective one of said digital cameras, said digital signal processors processing pixel data from pixel subsets of said digital cameras and generating pointer parameter data.

125. A camera based position tracking system according to claim 124, wherein said one processing stage creates a difference image.

126. A camera based position tracking system according to claim 125, wherein said one processing stage filters background features from said difference image.

127. A camera based position tracking system according to claim 126, wherein said one processing stage filters noise from said difference image.

128. A camera based position tracking system according to any one of claims 124 to 127, wherein a second of said processing stages includes a master digital signal processor receiving said pointer data from said digital signal processors, said master digital signal processor triangulating said pointer data.

129. A camera based position tracking system according to claim 128, wherein a third of said processing stages includes a computer receiving the location of said pointer from said master digital signal processor.

130. An imaging assembly comprising:

a substantially rectangular frame to surround a region of interest in front of a display surface; and

at least two digital cameras, each digital camera having a selectable pixel array and being mounted adjacent a different corner of said frame, each digital camera having a field of view that (i) images at least a portion of said display surface, (ii) images a least a portion of said frame, and (iii) extends at least beyond an adjacent peripheral edge of said frame, said digital cameras being oriented to capture overlapping images of said region of interest, each digital camera further comprising a processor processing a subset of pixel data of images captured by the digital camera to detect the existence of a pointer therein, wherein during processing of each captured image, said processor forms a difference image representing differences between pixel data in the captured image and corresponding pixel data in a background image, generates a histogram comprising a plurality of elements by selectively summing pixel intensities of said difference image, compares the elements of the histogram to a noise threshold and determines the existence of a pointer based on the result of said comparing.

131. An imaging assembly according to claim 130, wherein each digital camera comprises a CMOS digital camera.

132. An imaging assembly according to claim 130 or 131, wherein imaging components and the processor of each digital camera are mounted on a common board.

133. An imaging assembly according to claim 132, wherein a subset of pixels in the selectable pixel array of each digital camera provides the subset of pixel data to said associated processor.

134. An imaging assembly according to claim 133, wherein each subset of pixels includes contiguous rows of pixels.

135. An imaging assembly according to claim 134, wherein the subset of pixels of each digital camera pixel array is aligned so that the subset of pixels looks generally along a plane substantially coplanar to said display surface.

136. An imaging assembly according to claim 135, wherein the row of pixels in the subset of pixels of each digital camera pixel array, that acquires pixel data corresponding to a tip of said pointer within said region of interest, is determined to enable the associated processor to detect pointer contact and pointer hover over said display surface.

137. An imaging assembly according to claim 130 wherein said processor determines the existence of the pointer when a plurality of elements of the histogram exceed the noise threshold.

138. An imaging assembly according to claim 137 wherein said processor further determines the center of the pointer by determining the location of the element of the plurality having the highest value.

139. An imaging assembly according to claim 138 wherein said processor sums the intensities of the pixels of each column of the difference image to form the histogram.

140. A camera-based touch system comprising:

at least two image sensors associated with a touch surface and having overlapping fields of view encompassing said touch surface, each image sensor being oriented so that the field of view thereof (i) encompasses at least a portion of said touch surface, and (ii) extends at least beyond an adjacent peripheral edge of said touch surface, said at least two image sensors acquiring images of said touch surface from different vantages; and

processing structure processing pixel subsets of the images acquired by said at least two image sensors, pixel data of each pixel subset being processed to determine the existence of a pointer by calculating a median location of the pointer, said processing structure using the calculated median locations to triangulate the location of the pointer relative to said touch surface.

141. A touch system according to claim 140 wherein said at least two image sensors have fields of view looking generally along the plane of said touch surface.

142. A touch system according to claim 141 wherein pixel intensities of said pixel subsets are processed to determine the existence of the pointer.

143. A touch system according to any one of claims 140 to 142 wherein each of said image sensors communicates with an associated digital signal processor, said digital signal processor receiving image output from said image sensor and executing a find pointer routine to determine the median location of said pointer.

144. A touch system according to claim 143 wherein during said find pointer routine, said digital signal processor builds a vertical intensity histogram including columns of pixel intensities representing differences between an acquired image and a background image, the column of said vertical intensity histogram having the largest pixel intensity above a threshold level being used to define the center of said pixel subset, the width of said pixel subset being defined by columns of said vertical intensity histogram on opposite sides of the column defining the center of said pixel subset that have pixel intensities greater than said threshold level.

145. A touch system according to claim 144 wherein the digital signal processor associated with each image sensor analyses the pixels in said pixel subset to locate the pixel row where the pointer tip is located.

146. A touch system according to claim 145 wherein the digital signal processor associated with each image sensor creates a binary mask in said pixel subset with pixels of one value representing said pointer and pixels of a different value representing background to enable said median location and pointer tip location to be calculated.

147. A touch system according to claim 146 wherein the digital signal processor associated with each image sensor further executes an update background image routine to update the background image after each image is acquired.

148. A touch system according to claim 147 wherein during said update background image routine, the digital signal processor associated with each image sensor uses the equation:

$$B_{n+1}(i,j) = (1-a)\,B_n(i,j) + aI(i,j)$$

where:

B_{n+1} is the new background image;

B_n is the current background image;

I is the current acquired image;

i,j are the row and column coordinates of the background image pixels being updated; and

a is a number between 0 and 1 that indicates the degree of learning that should be taken from the current acquired image I.

149. A touch system according to claim 147 or 148 wherein the digital signal processor associated with each image sensor further determines the differences between each acquired image and the background image to detect changing light conditions.

150. A touch system according to claim 149 wherein said processing structure adjusts the exposure of each image sensor.

151. A touch system according to any one of claims 141 to 150 wherein said processing structure further executes a touch surface determination routine to calculate the orientation of the touch surface as seen by each image sensor.

152. A touch system according to any one of claims 141 to 151 wherein said processing structure further calculates the velocity of a pointer as said pointer moves toward said touch surface within the fields of view of said image sensors.

153. A touch system according to claim 152 wherein said processing structure tracks said pointer within the fields of view of said image sensors.

154. A touch system according to claim 153 wherein said processing structure tracks said pointer using at least one Kalman filter.

155. A camera-based touch system comprising:

a generally rectangular passive touch surface on which contacts are made using a pointer;

at least two digital image sensors, each image sensor being mounted adjacent a different corner of said touch surface, said digital image sensors having overlapping fields of view encompassing said touch surface, the field of view of each digital image sensor extending at least beyond an adjacent peripheral edge of said touch surface, said digital image sensors acquiring images looking generally across said touch surface; and

processing structure receiving image data output by said digital image sensors, said processing structure processing the image data to determine the existence of a pointer in acquired images by calculating median locations x of the pointer and using the calculated median locations to triangulate the location of said pointer relative to said touch surface and to determine whether said pointer is in contact with said touch surface.

156. A touch system according to claim 155 wherein each of said image sensors communicates with an associated digital signal processor, said digital signal processor receiving image output from said image sensor and executing a find pointer routine to determine the median location x of said pointer.

157. A touch system according to claim 156 wherein during said find pointer routine, said digital signal processor of each digital camera builds a vertical intensity histogram including columns of pixel intensities representing differences between an acquired image and a background image, the column of said vertical intensity histogram having the largest pixel intensity above a threshold level being used to define the center of said pixel subset, the width of said pixel subset being defined by columns of said vertical intensity histogram on opposite sides of the column defining the center of said pixel subset that have pixel intensities greater than a threshold level.

158. A touch system according to claim 157 wherein the digital signal processor associated with each image sensor analyses the pixels in said pixel subset to locate the pixel row where the pointer tip is located and determine said pointer tip location z.

159. A touch system according to claim 158 wherein the digital signal processor associated with each image sensor creates a binary mask in said pixel subset with pixels of one value representing said pointer and pixels of a different value representing background to enable said median location x and pointer tip location z to be calculated.

160. A touch system according to any one of claims 156 to 159 wherein the digital signal processor associated with each image sensor further executes an update background image routine to update the background image after each image is acquired.

161. A touch system according to claim 160 wherein during said update background image routine, the digital signal processor associated with each image sensor uses the equation:

$$B_{n+1}(i,j) = (l-a)\, B_n(i,j) + aI(i,j)$$

where:

B_{n+1} is the new background image;

B_n is the current background image;

I is the current acquired image;

i,j are the row and column coordinates of the background image pixels being updated; and

a is a number between 0 and 1 that indicates the degree of learning that should be taken from the current acquired image I.

162. A touch system according to claim 160 or 161 wherein the digital signal processor associated with each image sensor further determines the differences between each acquired image and the background image to detect changing light conditions.

163. A touch system according to claim 162 wherein said processing structure adjusts the exposure of each image sensor.

164. A touch system according to any one of claims 155 to 163 wherein said processing structure further executes a touch surface determination routine to calculate the orientation of the touch surface as seen by each image sensor.

165. A touch system according to any one of claims 155 to 164 wherein said processing structure further calculates the velocity of a pointer as said pointer moves toward said touch surface within the fields of view of said image sensors.

166. A touch system according to claim 165 wherein said processing structure tracks said pointer within the fields of view of said digital cameras.

167. A touch system according to claim 166 wherein said processing structure tracks said pointer using at least one Kalman filter.

168. A touch system according to any one of claims 155 to 167 wherein said touch surface is bordered by a frame and wherein each of said image sensors is mounted to said frame via a frame assembly, each image sensor being oriented so that

the field of view thereof looks downward and generally along the plane of said touch surface.

169. A touch system according to any one of claims 155 to 168 further including a computer coupled to said processing structure, said computer receiving pointer location information from said processing structure and using said pointer location information to update an applications program executed thereby.

170. A touch system according to claim 169 wherein computer display information is presented on said touch surface.

171. A camera-based touch system comprising:

at least two cameras associated with a touch surface and having overlapping fields of view encompassing said touch surface, the field of view of each camera extending at least beyond an adjacent peripheral edge of said touch surface, said at least two cameras acquiring images of said touch surface from different locations and generating image data; and

processing means triangulating pointer data to determine the location of said pointer relative to said touch surface, wherein the at least two cameras further generate pixel subsets of image data from said image data and the processing means receives said pixel subsets of image data generated by the at least two cameras, the processing means processing said pixel subsets to generate the pointer data when a pointer exists in the said acquired images.

172. A touch system according to claim 171 wherein said at least two cameras are digital cameras having fields of view looking generally along the plane of said touch surface.

173. A touch system according to claim 172 wherein the pointer data includes a pointer median line x and a pointer tip location z.

174. A touch system according to claim 172 or 173 wherein each digital camera comprises a CMOS digital camera having a selectable pixel array.

175. A touch system according to claim 173 or 174 wherein each digital camera includes a CMOS image sensor and a digital signal processor, said digital signal processor receiving image output from said image sensor and executing a find pointer routine to determine if a pointer is in each image acquired by said digital camera, and if so the median line x of said pointer, the digital signal processor of each digital camera providing pointer data to a master digital signal processor, said master digital signal processor triangulating the pointer data to determine the pointer location.

176. A touch system according to claim 175 wherein the digital signal processor of each digital camera analyses the pixel data to locate the pixel row of the selected pixel subset where the pointer tip is located thereby to determine said pointer tip location z.

177. A touch system according to claim 173 or 174 wherein for image data received from each digital camera, said processing means calculates an angle ϕ_{cam} using the equation:

$$\tan \phi_{cam} = \frac{2\left(\frac{x}{a}\right)\tan\frac{FOV}{2}}{1-\left(2\frac{x}{a}-1\right)\tan^2\frac{FOV}{2}}$$

where:

x is the number representing the median line or tip of the pointer; and

a is the total length enclosed by the field of view (FOV) of the digital camera at a distance from the camera;

said processing means using the calculated angles to determine the pointer location relative to said touch surface.

178. A touch system according to claim 177 wherein said calculated angles are adjusted to take into account digital camera offsets prior to determination of said pointer location.

179. A touch system according to claim 177 or 178 comprising at least three digital cameras, said processing means determining the pointer location using triangulation for multiple pairs of digital cameras and averaging the determined pointer locations.

180. A touch system according to any one of claims 171 to 179 wherein said processing means further executes a touch surface determination routine to calculate the orientation of the touch surface as seen by each digital camera to determine when the pointer is in contact with said touch surface and when said pointer is hovering above said touch surface.

181. A touch system according to any one of claims 171 to 174, wherein said touch surface is substantially rectangular wherein a camera is positioned adjacent each corner of said touch surface.

182. A touch system according to claim 181, wherein each selected pixel subset includes contiguous rows of pixels, and wherein pixel data acquired by said contiguous rows of pixels is processed at full resolution.

183. A touch system according to any one of claims 171 to 173, comprising more than two cameras having overlapping fields of view and wherein said processing means triangulates pointer data from multiple pairs of digital cameras to determine the location of the pointer relative to said touch surface.

184. A touch system according to claim 183, wherein said processing means includes a plurality of processing stages, said processing stages processing image data in a manner to stage bandwidth.

185. A touch system according to claim 184, wherein one of said processing stages includes a plurality of digital signal processors each associated with a respective one of said cameras, said digital signal processors processing pixel data from said selected pixel subsets and generating said pointer data.

186. A touch system according to claim 185, wherein a second of said processing stages includes a master digital signal processor receiving said pointer data from said digital signal processors, said master digital signal processor triangulating said pointer data to determine said pointer location.

187. A touch system according to any one of claims 183 to 186, wherein an image is projected on said touch surface, wherein each of said cameras includes a pixel array proportional to the number of pixels in said projected image, and wherein said processing means is able to resolve the location of said pointer relative to said touch surface with sufficient accuracy with respect to the pixels in said projected image.

188. A touch system according to any one of claims 171 to 187 further comprising a computer coupled to said processing means, said computer receiving said pointer location from said processing means.

190. A method of detecting the position of a pointer relative to a touch surface comprising the steps of:

acquiring multiple overlapping images of a pointer relative to said touch surface using digital cameras having overlapping fields of view, the field of view of each digital camera at least extending beyond an adjacent peripheral edge of said touch surface;

selecting pixel subsets of said acquired images; and

processing pixel data of said pixel subsets to detect the existence of said pointer therein and to determine the location of said pointer relative to said touch surface using triangulation.

191. The method of claim 190, wherein during said processing step, the pixel data is processed to determine when said pointer is in contact with said touch surface and when said pointer is hovering over said touch surface.

192. The method of claim 191, wherein during said processing step the existence of said pointer is determined by calculating median lines of the pointer and wherein the location of said pointer is determined by calculating the intersection point of median lines and using triangulation to determine the coordinates of said intersection point.

193. The method of claim 192 wherein during said processing step a difference image of said acquired images is created.

194. The method of claim 193 wherein during said processing step background features are filtered from said difference image.

195. The method of claim 193 wherein during said processing step noise is filtered from said difference image.

196. The method of any one of claims 190 to 195 wherein said processing further includes the step of tracking the pointer as the pointer approaches the touch surface.

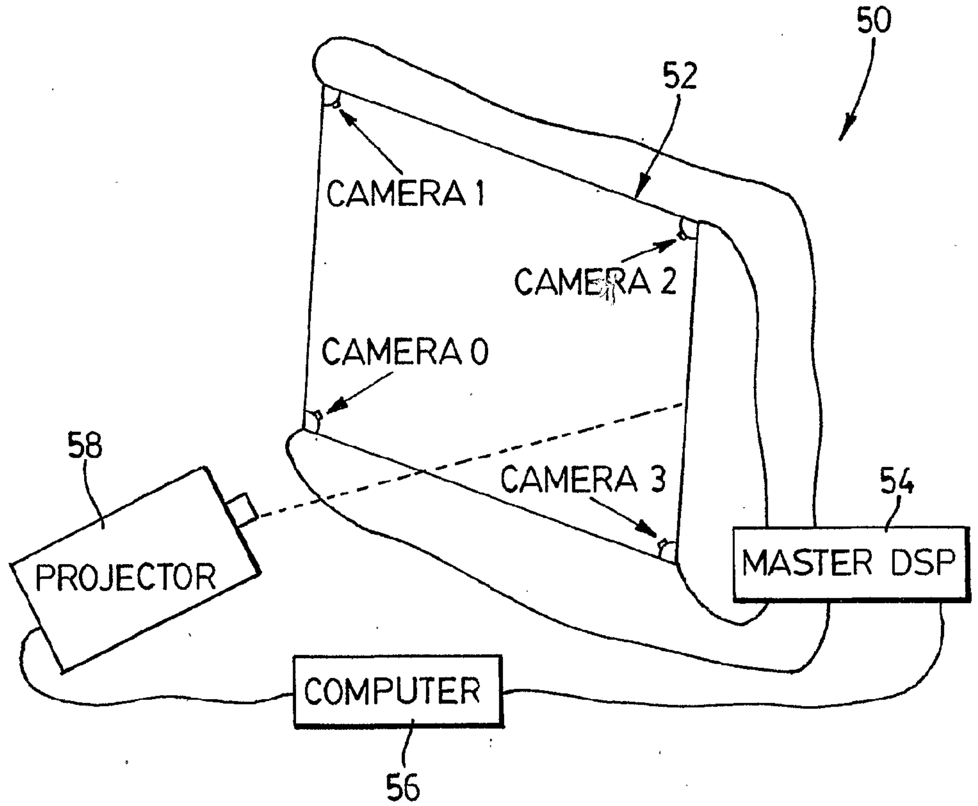

50
52
CAMERA 1
CAMERA 2
CAMERA 0
58
CAMERA 3
54
MASTER DSP
PROJECTOR
COMPUTER
56

FIG. 1

2/11

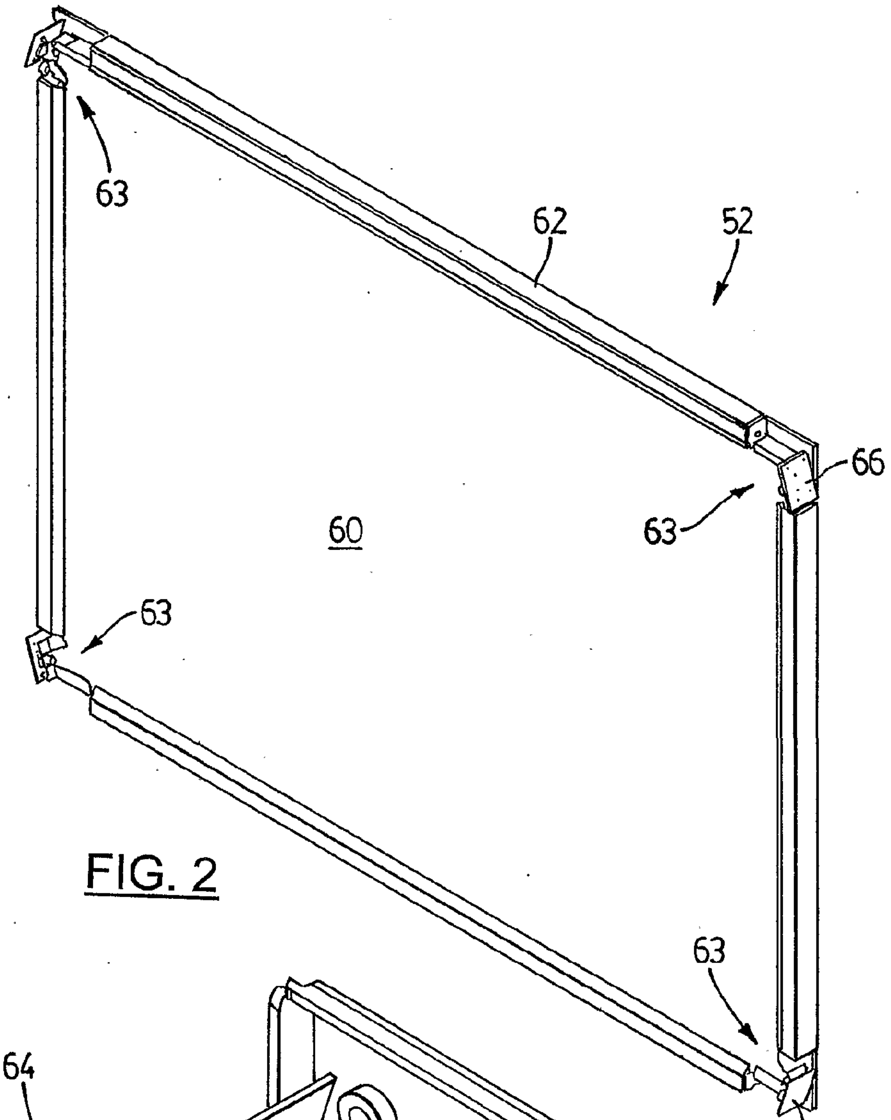


FIG. 2

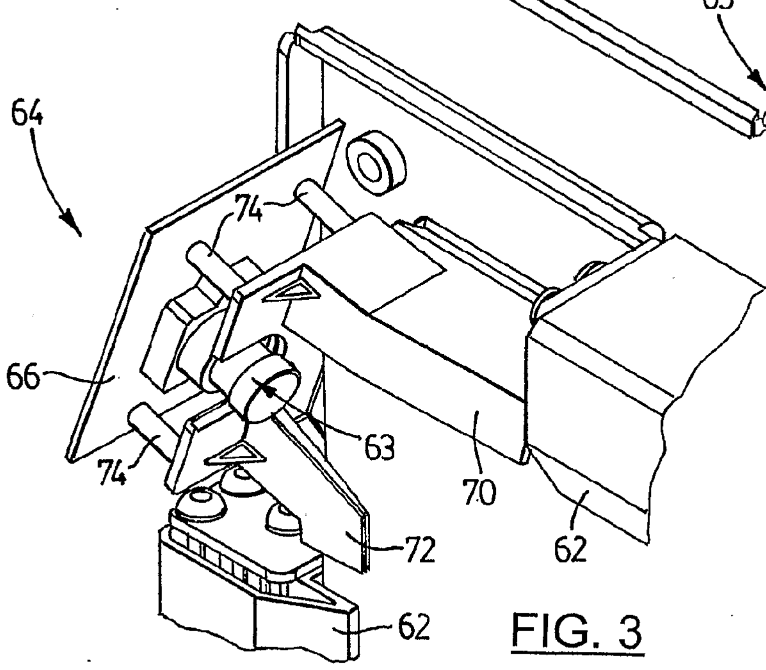


FIG. 3

3/11

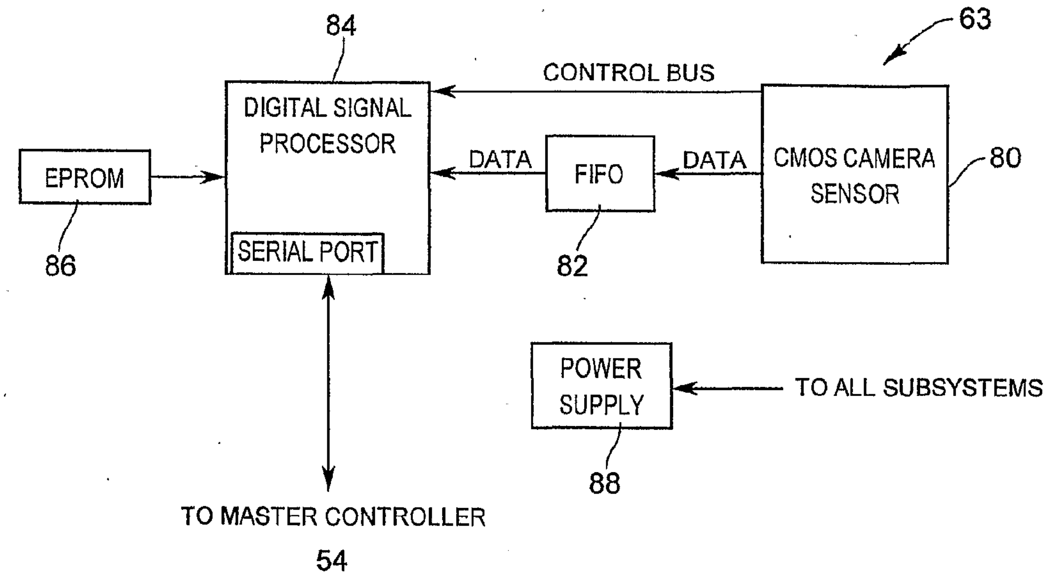


FIG. 4

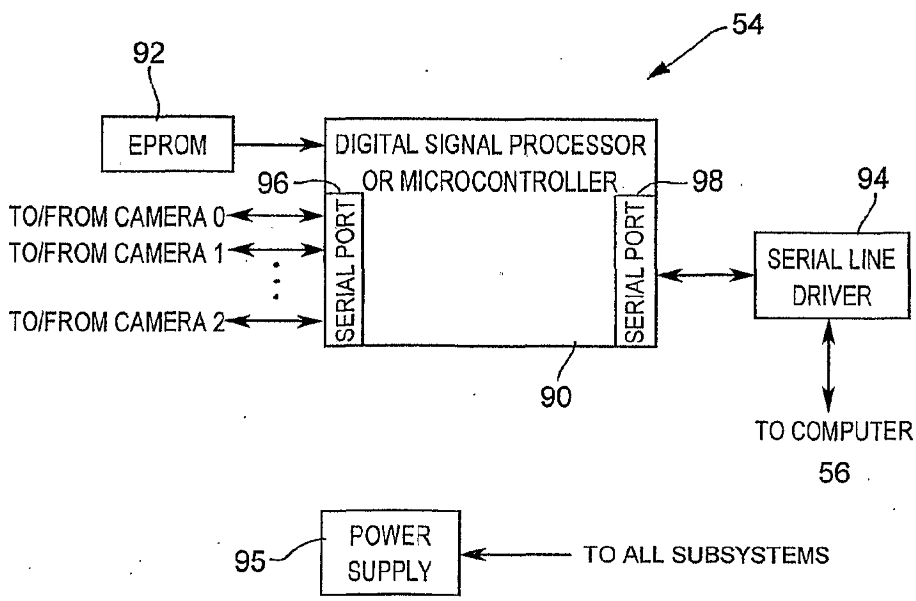


FIG. 5

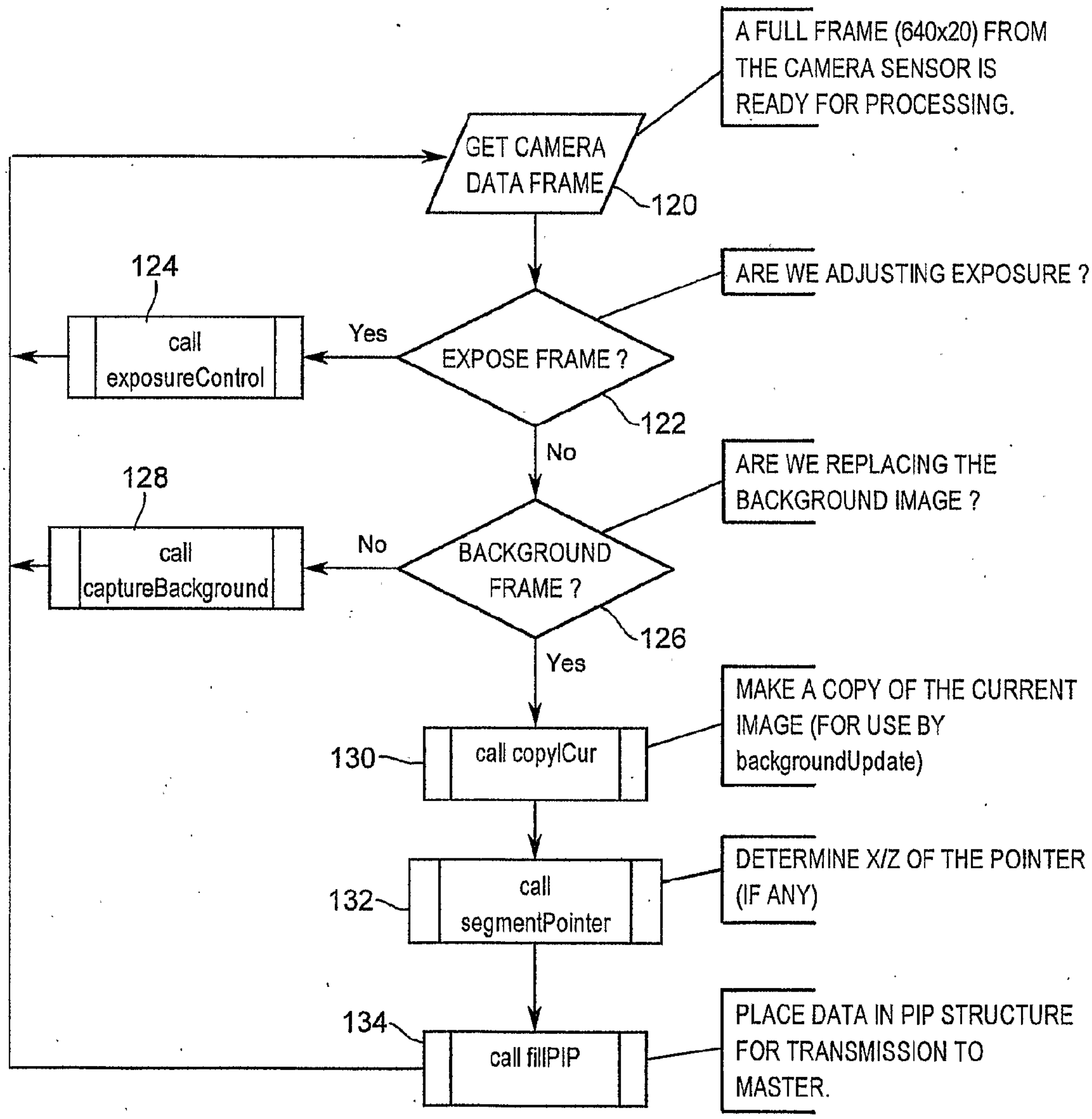

A FULL FRAME (640x20) FROM THE CAMERA SENSOR IS READY FOR PROCESSING.
GET CAMERA DATA FRAME
120
ARE WE ADJUSTING EXPOSURE ?
124
call exposureControl
Yes
EXPOSE FRAME ?
122
No
ARE WE REPLACING THE BACKGROUND IMAGE ?
128
call captureBackground
No
BACKGROUND FRAME ?
126
Yes
MAKE A COPY OF THE CURRENT IMAGE (FOR USE BY backgroundUpdate)
130
call copyICur
DETERMINE X/Z OF THE POINTER (IF ANY)
132
call segmentPointer
134
call fillPIP
PLACE DATA IN PIP STRUCTURE FOR TRANSMISSION TO MASTER.

FIG. 6

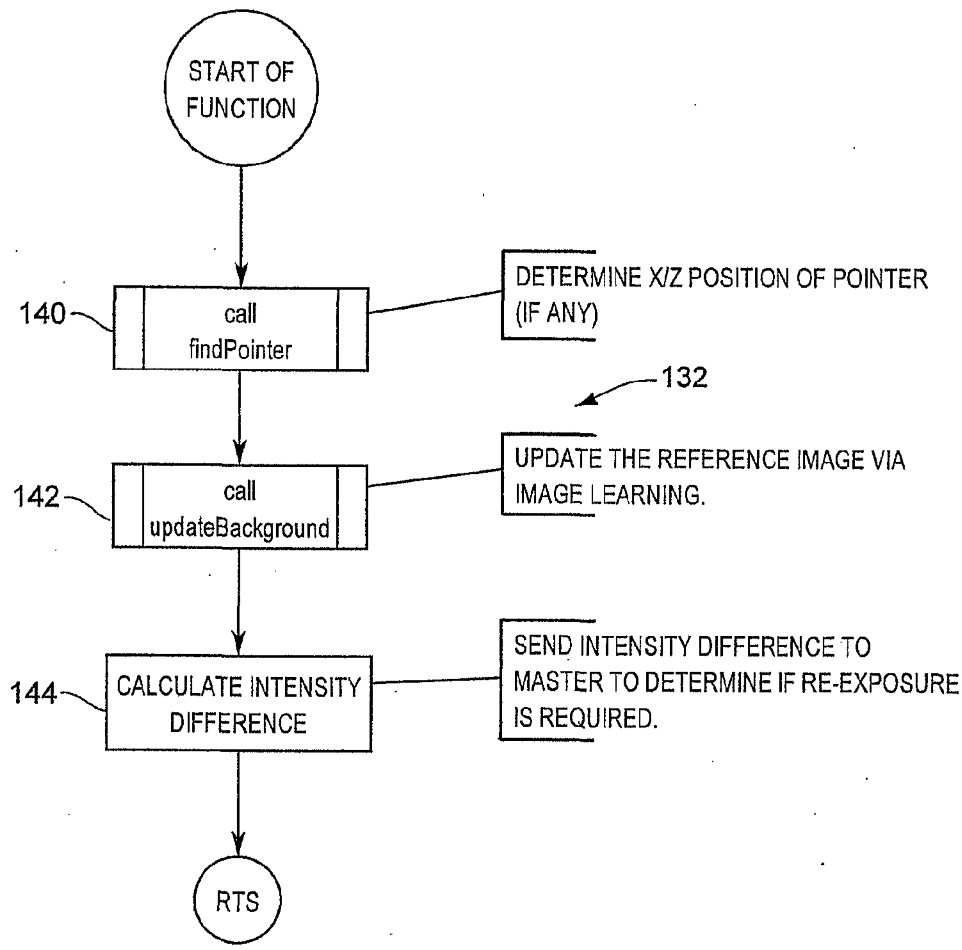

START OF FUNCTION
140
call findPointer
DETERMINE X/Z POSITION OF POINTER (IF ANY)
132
142
call updateBackground
UPDATE THE REFERENCE IMAGE VIA IMAGE LEARNING.
144
CALCULATE INTENSITY DIFFERENCE
SEND INTENSITY DIFFERENCE TO MASTER TO DETERMINE IF RE-EXPOSURE IS REQUIRED.
RTS

FIG. 7

6/11

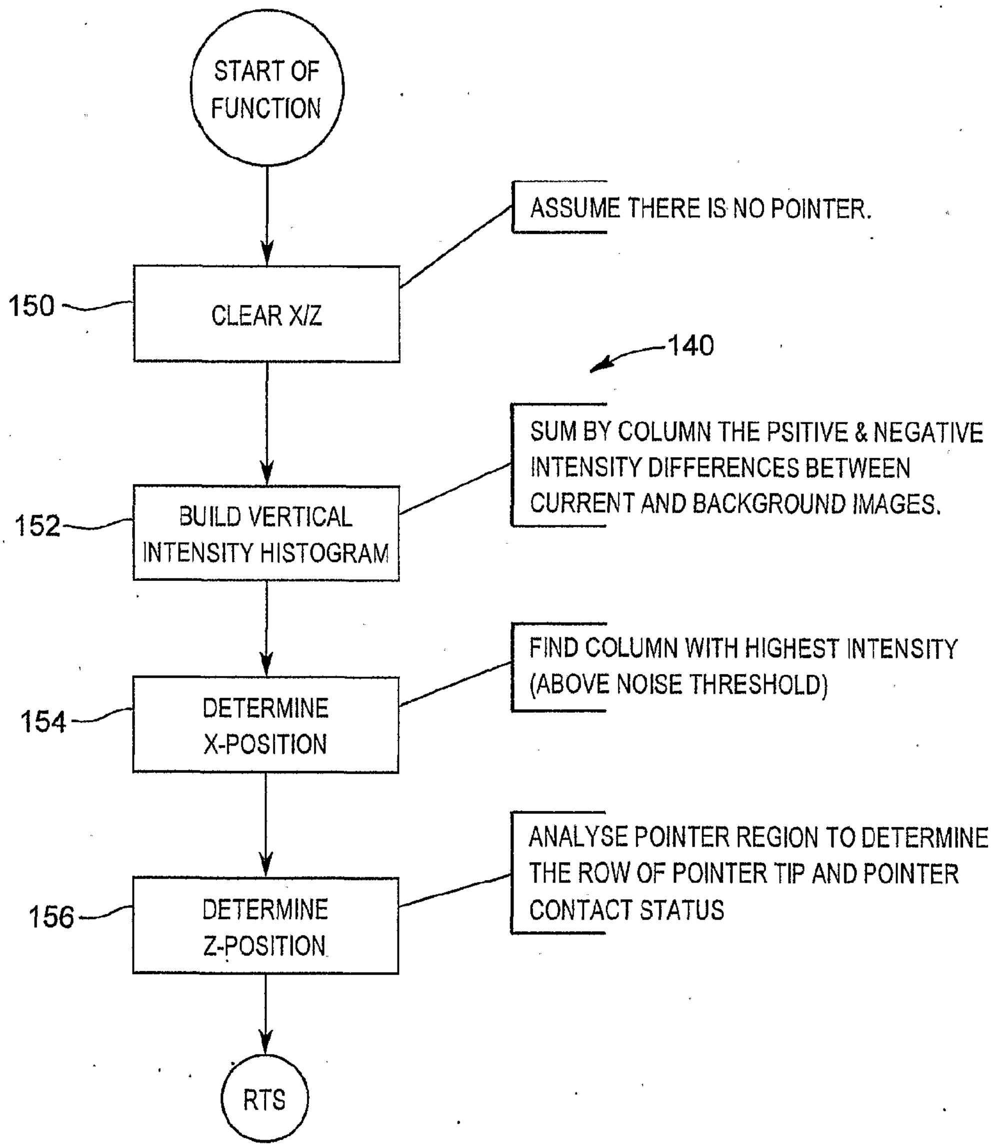

START OF FUNCTION
ASSUME THERE IS NO POINTER.
150
CLEAR X/Z
140
SUM BY COLUMN THE PSITIVE & NEGATIVE INTENSITY DIFFERENCES BETWEEN CURRENT AND BACKGROUND IMAGES.
152
BUILD VERTICAL INTENSITY HISTOGRAM
FIND COLUMN WITH HIGHEST INTENSITY (ABOVE NOISE THRESHOLD)
154
DETERMINE X-POSITION
ANALYSE POINTER REGION TO DETERMINE THE ROW OF POINTER TIP AND POINTER CONTACT STATUS
156
DETERMINE Z-POSITION
RTS

FIG. 8

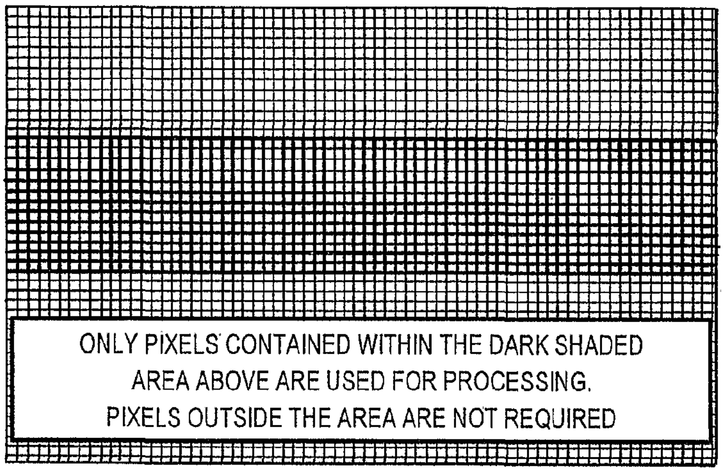

ONLY PIXELS CONTAINED WITHIN THE DARK SHADED
AREA ABOVE ARE USED FOR PROCESSING.
PIXELS OUTSIDE THE AREA ARE NOT REQUIRED

FIG. 9

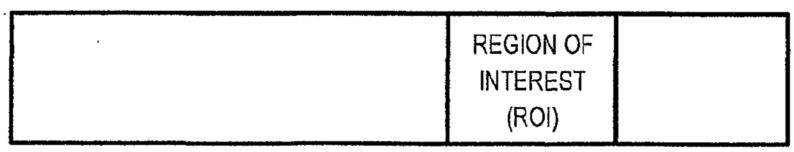

REGION OF
INTEREST
(ROI)

FIG. 10

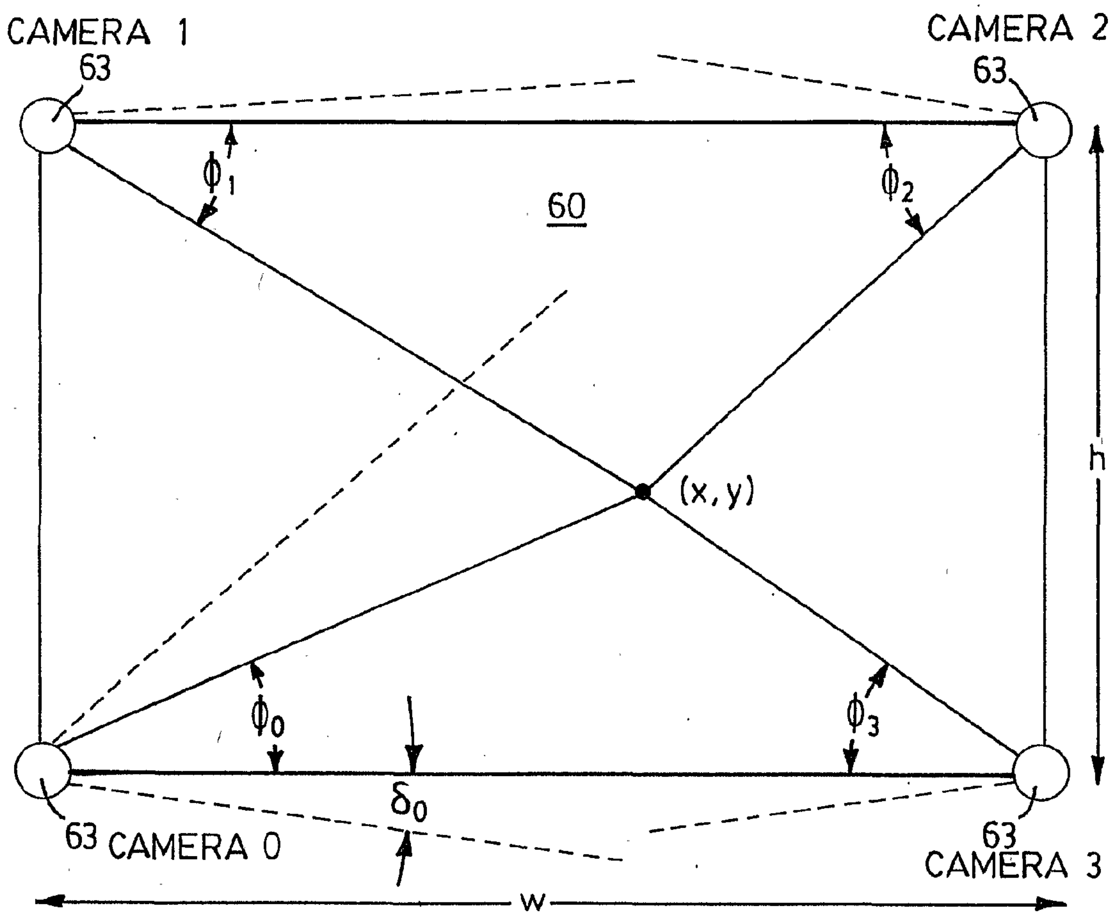

CAMERA 1
63
CAMERA 2
63
ϕ_1
60
ϕ_2
h
(x, y)
ϕ_0
ϕ_3
δ_0
63 CAMERA 0
63
CAMERA 3
w

FIG. 11

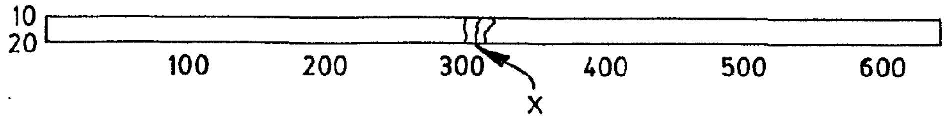
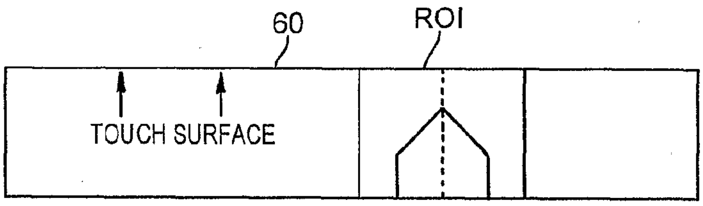
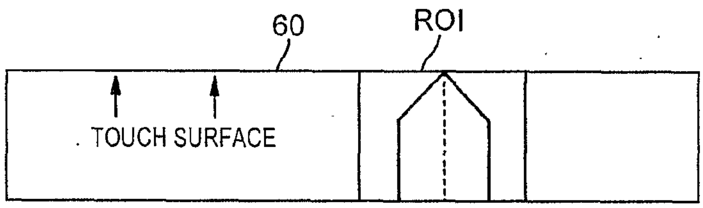

10
20
100
200
300
400
500
600
x

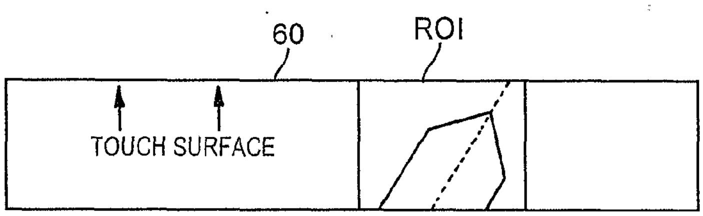
FIG. 12



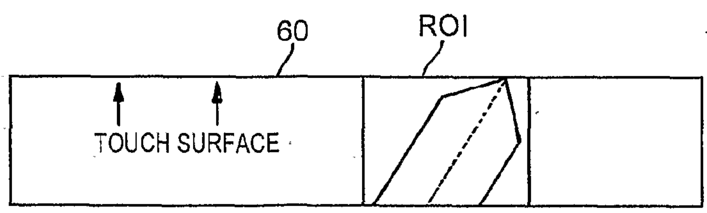
PERPENDICULAR POINTER APPROACHING TOUCH SURFACE
CALCULATED CENTER LINE SHOWS EXPECTED POINT OF CONTACT.



PERPENDICULAR POINTER IN CONTACT WITH TOUCH SURFACE



NON-PERPENDICULAR POINTER APPROACHING TOUCH SURFACE
CALCULATED CENTER LINE SHOWS EXPECTED POINT OF CONTACT.



NON-PERPENDICULAR POINTER IN CONTACT WITH TOUCH SURFACE.

FIG. 13

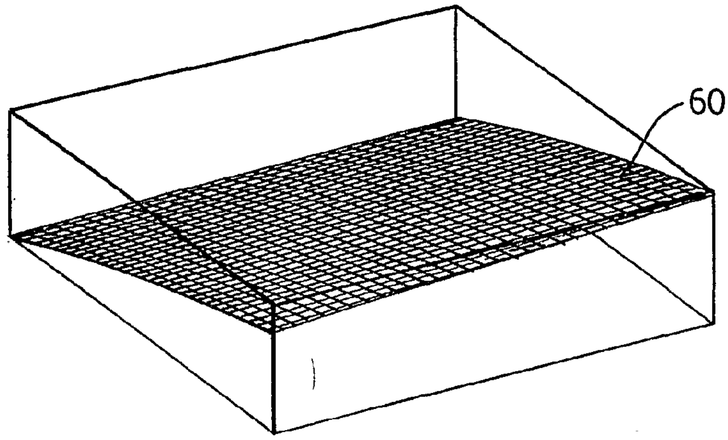


FIG. 14

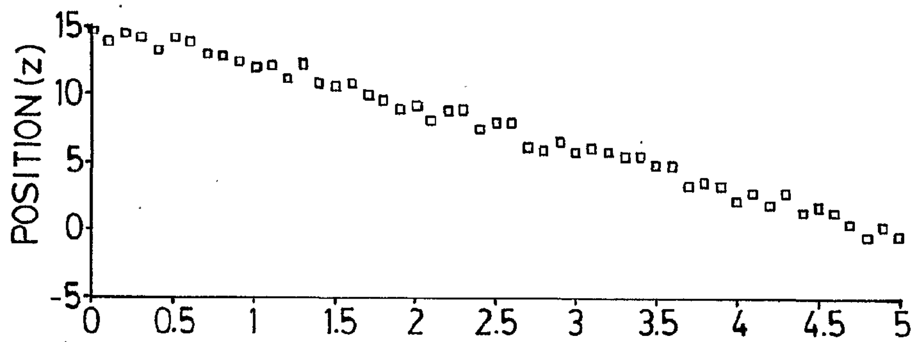


FIG. 15

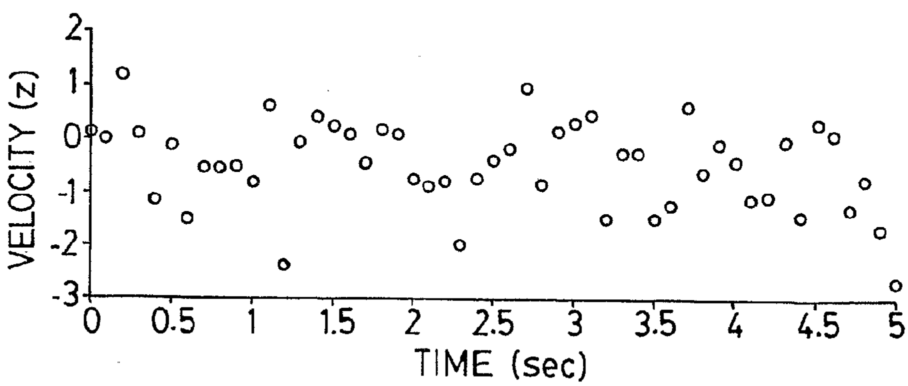


FIG. 16

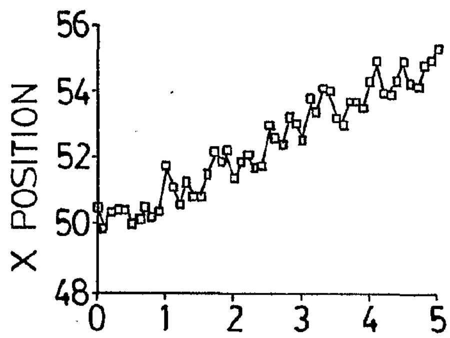


FIG. 17a

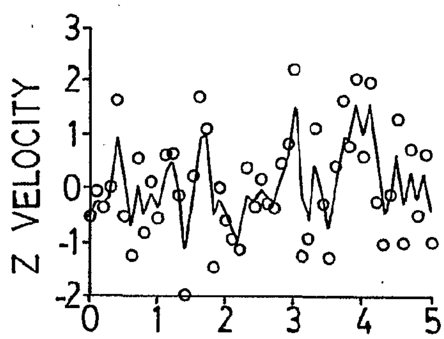


FIG. 17b

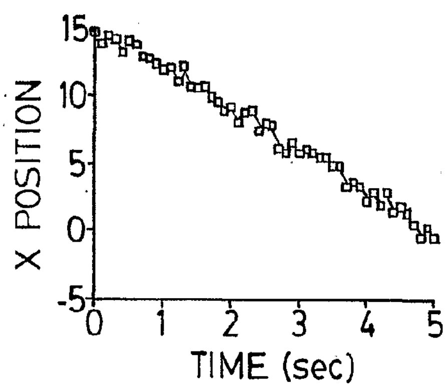


FIG. 17c

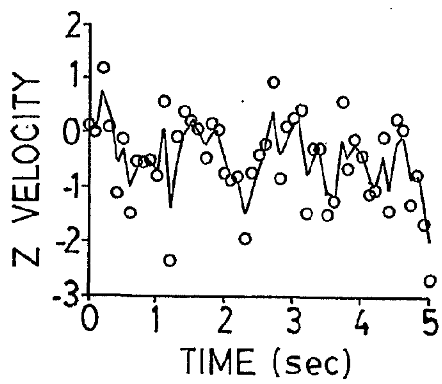
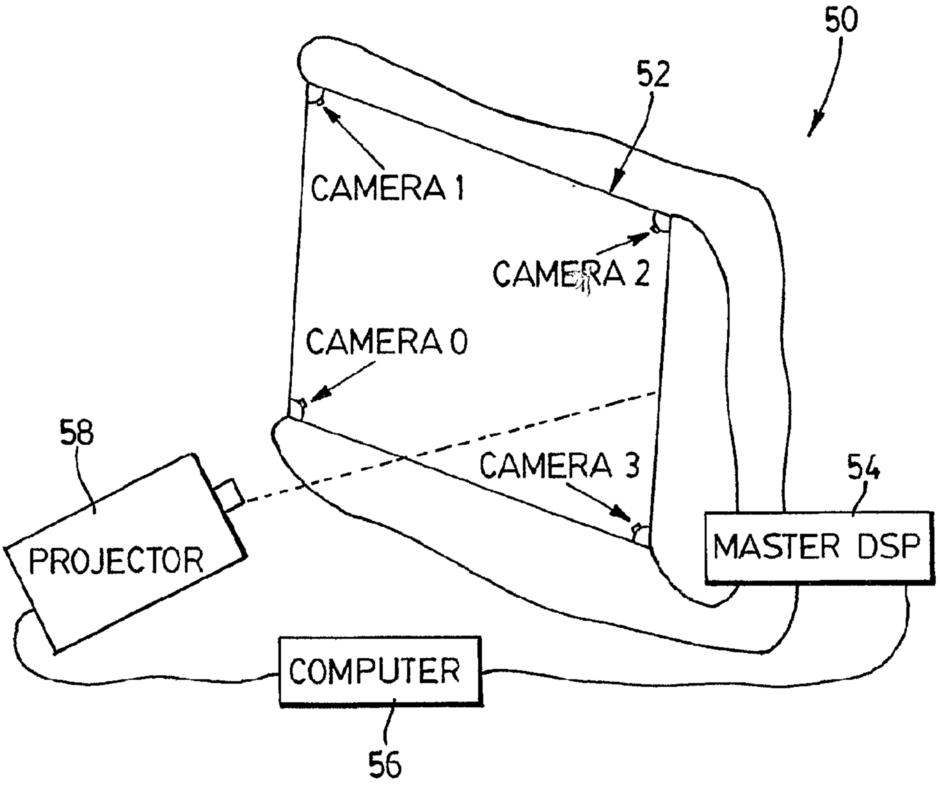


FIG. 17d


50
52
CAMERA 1
CAMERA 2
CAMERA 0
58
CAMERA 3
54
MASTER DSP
PROJECTOR
COMPUTER
56