



US 20060238549A1

(19) **United States**

(12) **Patent Application Publication**
Marks

(10) **Pub. No.: US 2006/0238549 A1**

(43) **Pub. Date: Oct. 26, 2006**

(54) **SYSTEM AND METHOD FOR OBJECT TRACKING**

Publication Classification

(75) Inventor: **Richard L. Marks**, Foster City, CA (US)

(51) **Int. Cl.**
G09G 5/00 (2006.01)
(52) **U.S. Cl.** **345/633**

Correspondence Address:
CARR & FERRELL LLP
2200 GENG ROAD
PALO ALTO, CA 94303 (US)

(57) **ABSTRACT**

(73) Assignee: **Sony Computer Entertainment Inc.**

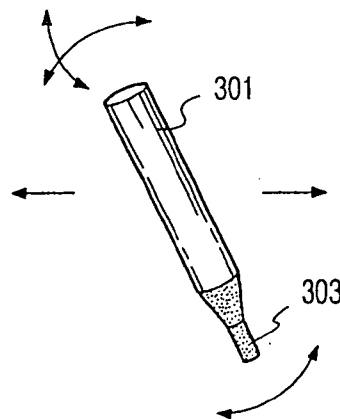
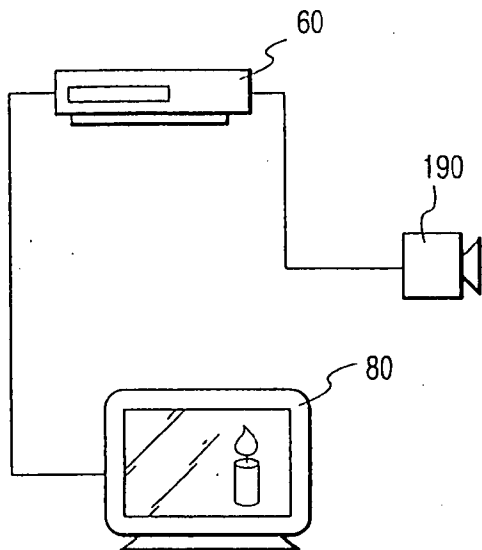
A system for tracking an object is disclosed. The exemplary tracking system comprises an input device configured to detect two-dimensional input pixel data from a prop device and a multiprocessor unit configured to calculate three-dimensional position and orientation data associated with the prop device from the two-dimensional input pixel data. An exemplary method for tracking an object is also disclosed. Through this exemplary method, pixel data is received from an input device and edges of an object are defined. Three-dimensional position and orientation data of the object are calculated, wherein the edges of the object are associated with the three-dimensional position and orientation data of the prop device.

(21) Appl. No.: **11/448,454**

(22) Filed: **Jun. 6, 2006**

Related U.S. Application Data

(63) Continuation of application No. 10/928,778, filed on Aug. 26, 2004, which is a continuation of application No. 09/621,578, filed on Jul. 21, 2000, now Pat. No. 6,795,068.



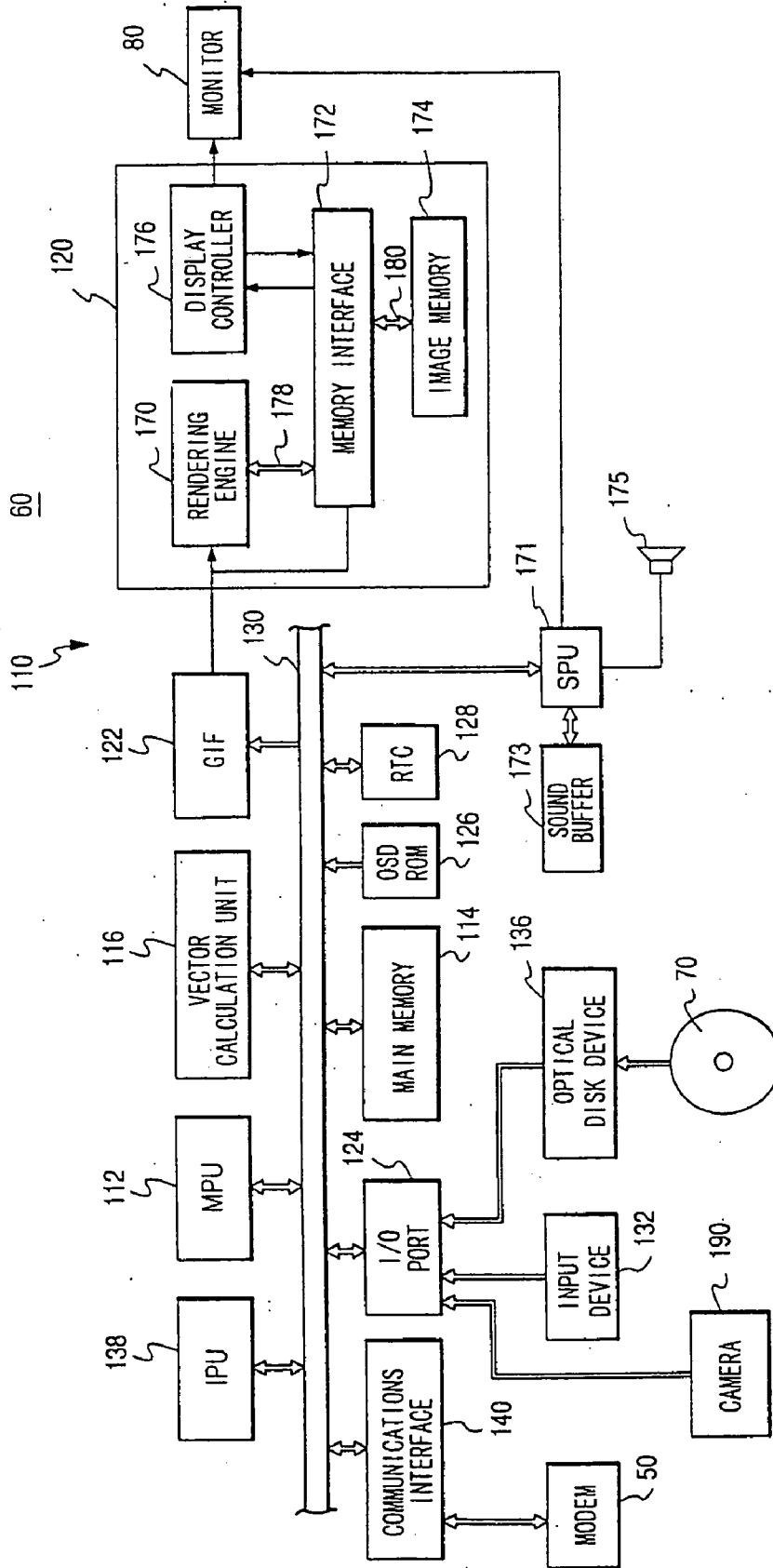
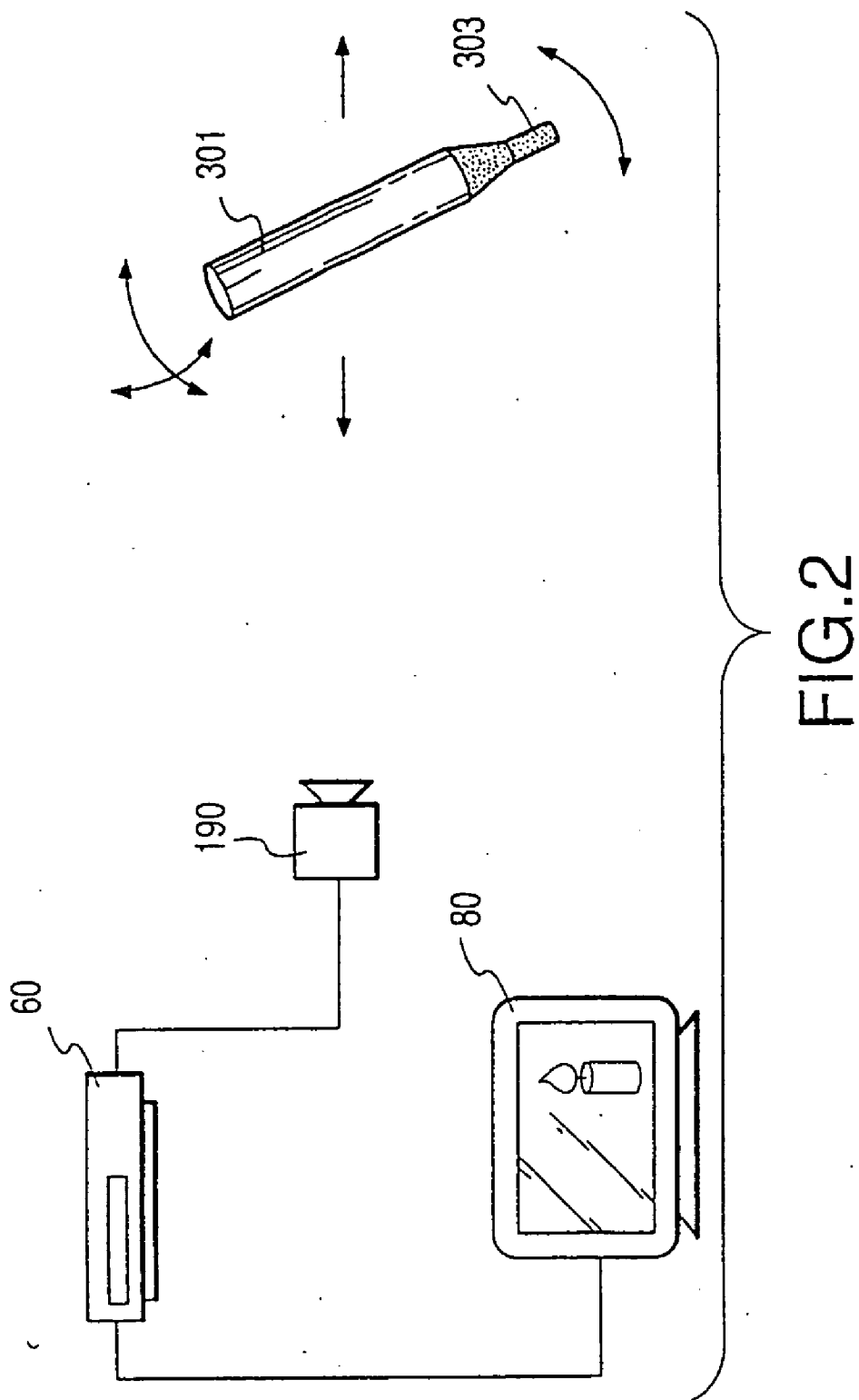


FIG.1



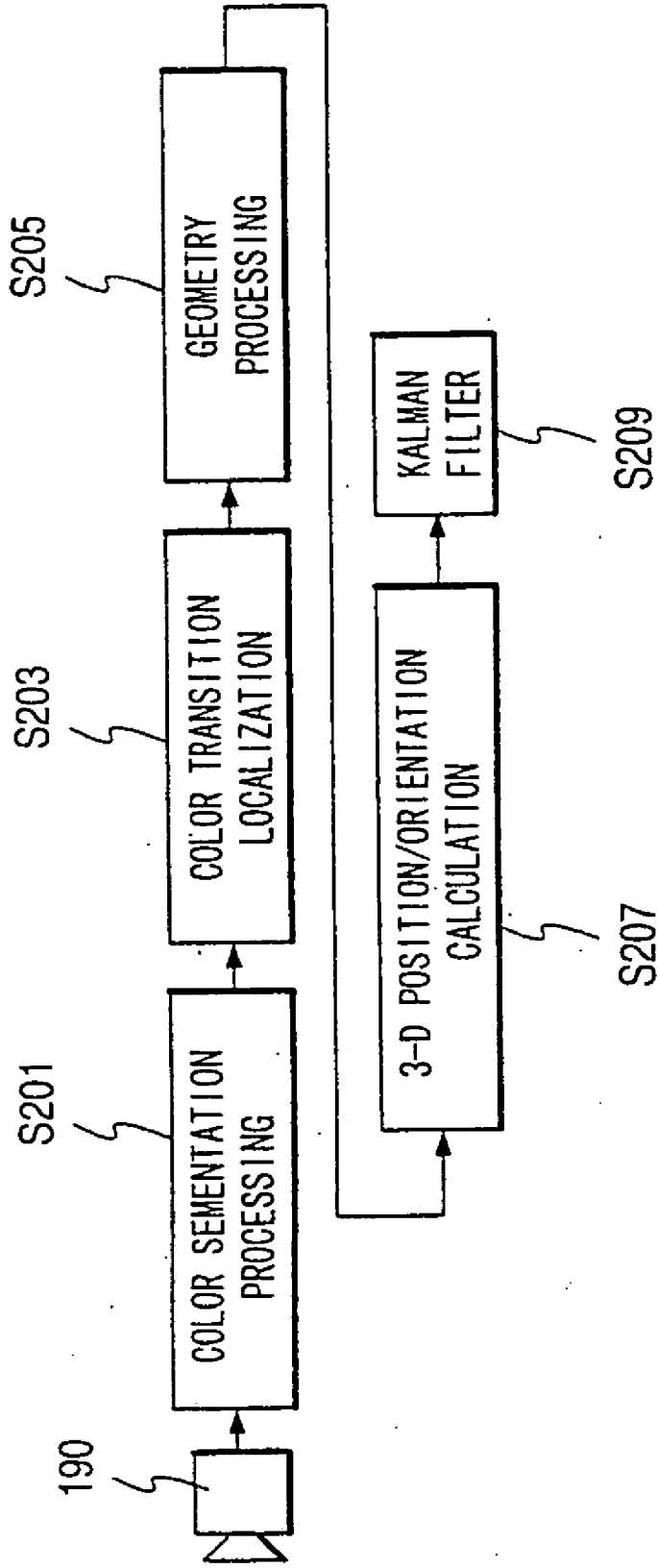


FIG.3

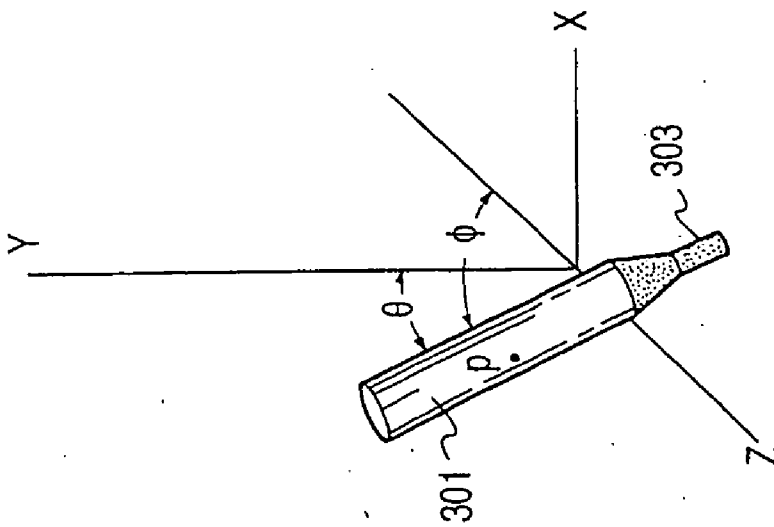


FIG. 4A

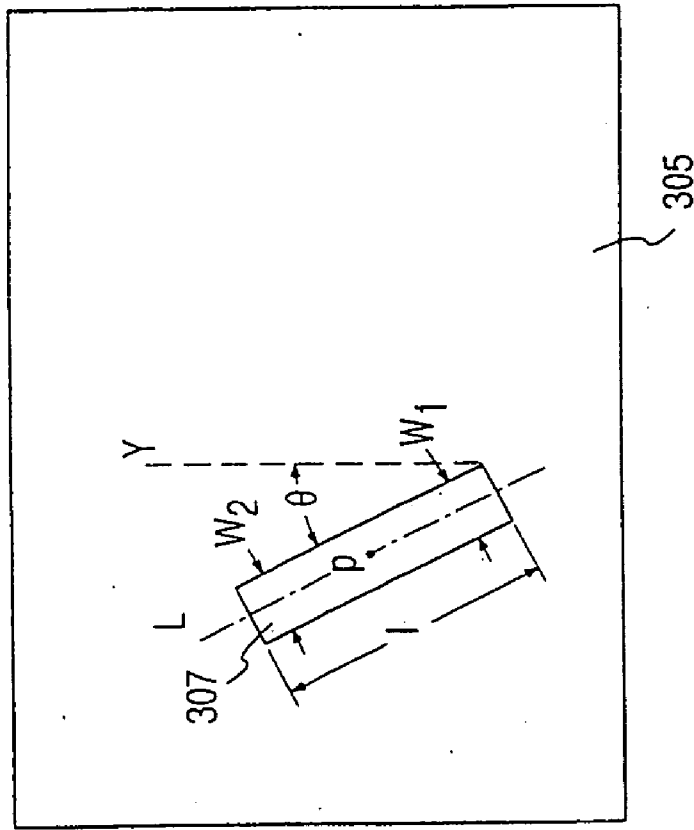


FIG. 4B

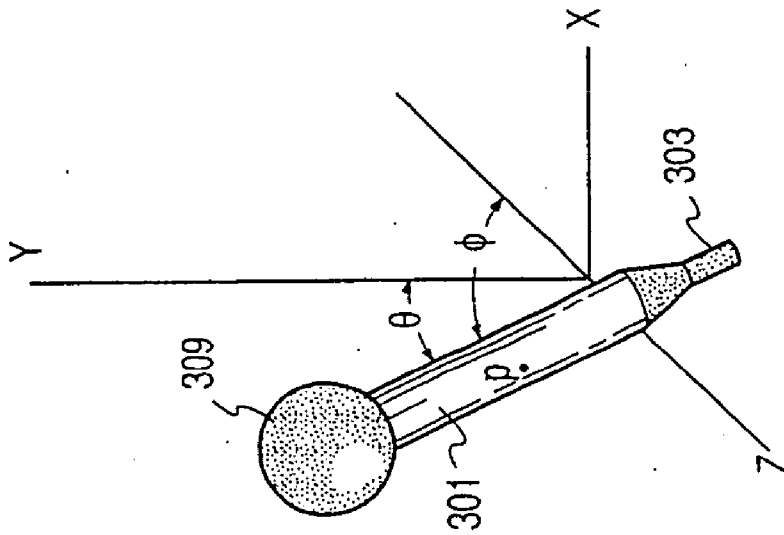


FIG. 5A

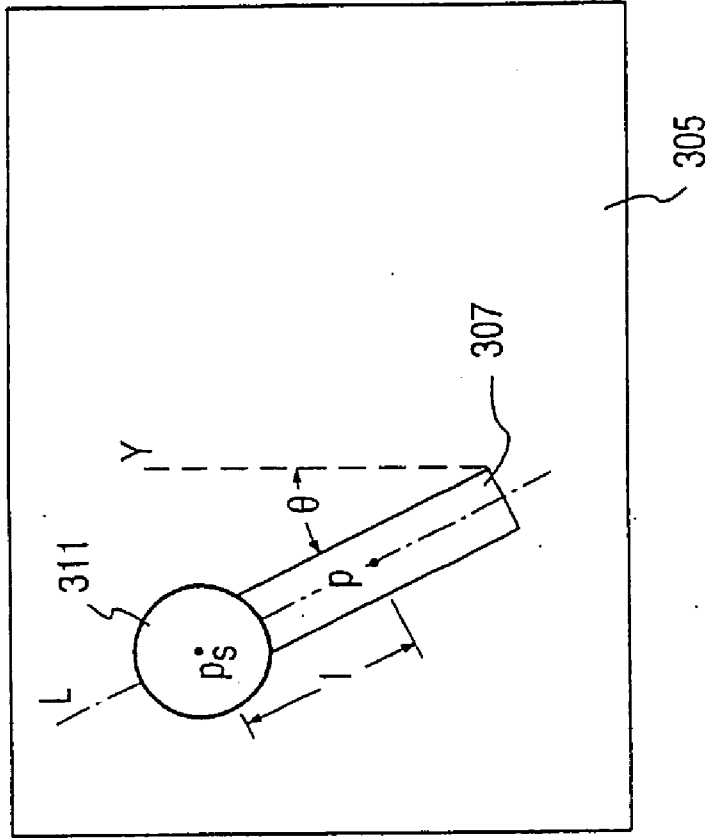


FIG. 5B

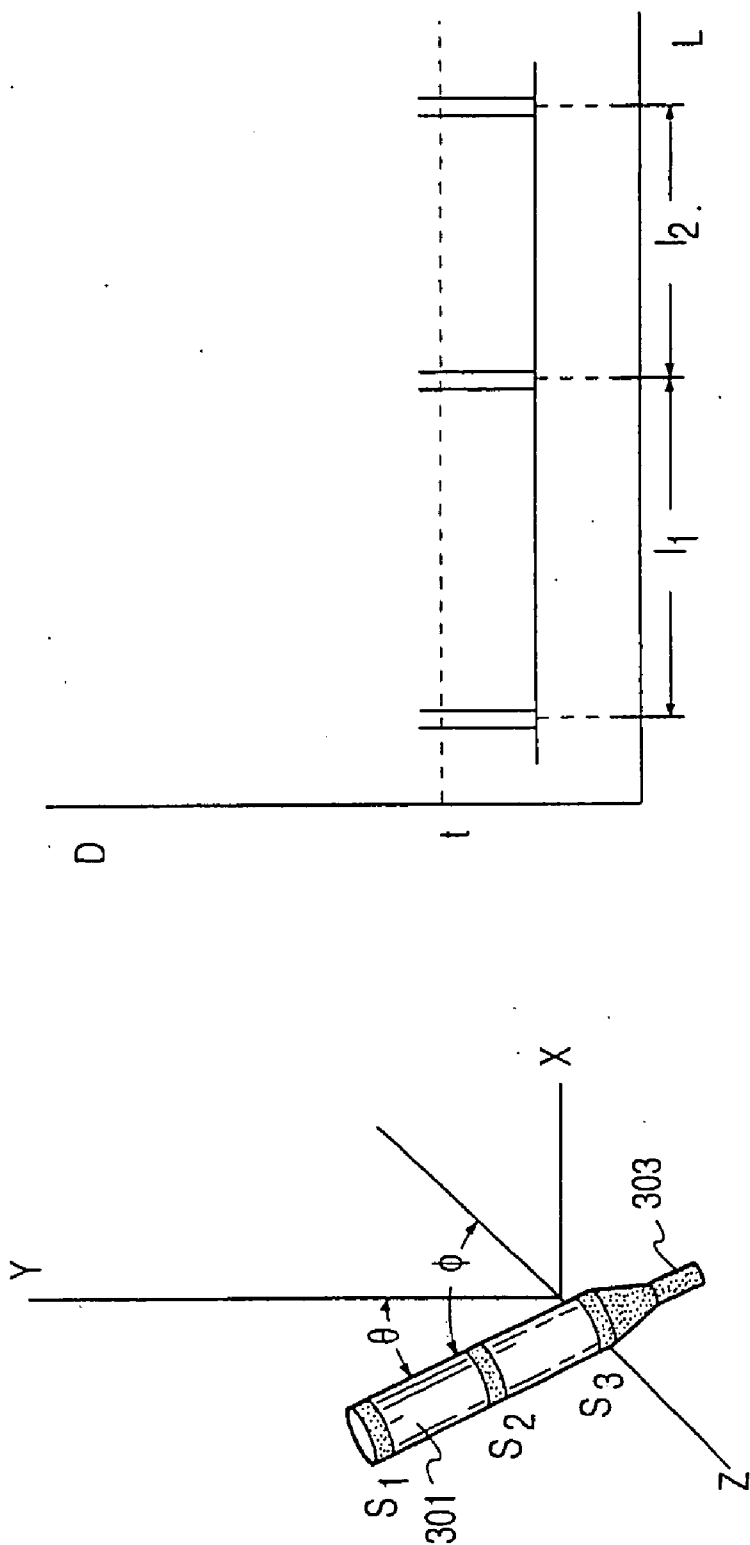


FIG. 6B

FIG. 6A

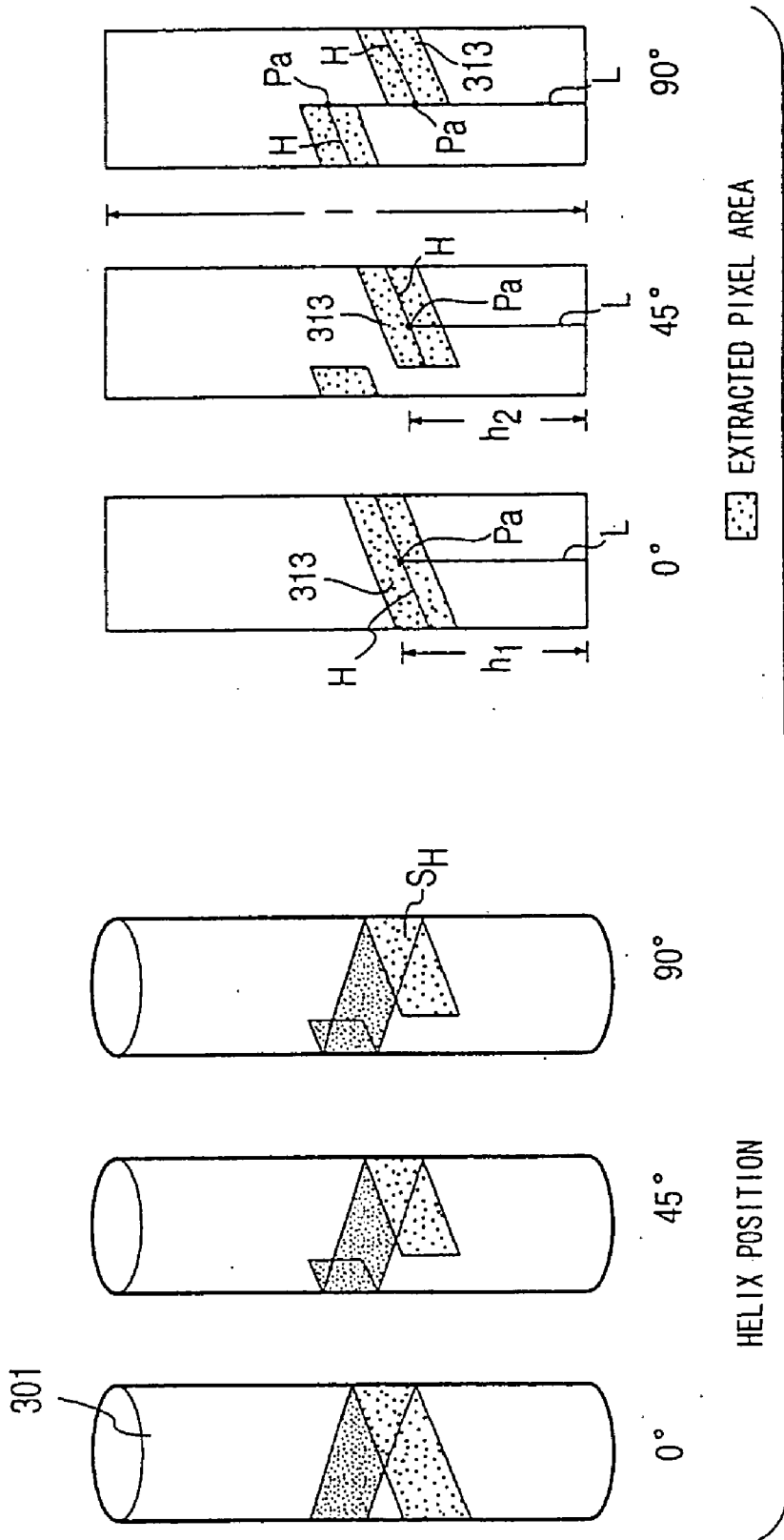


FIG.7

FIG.8A

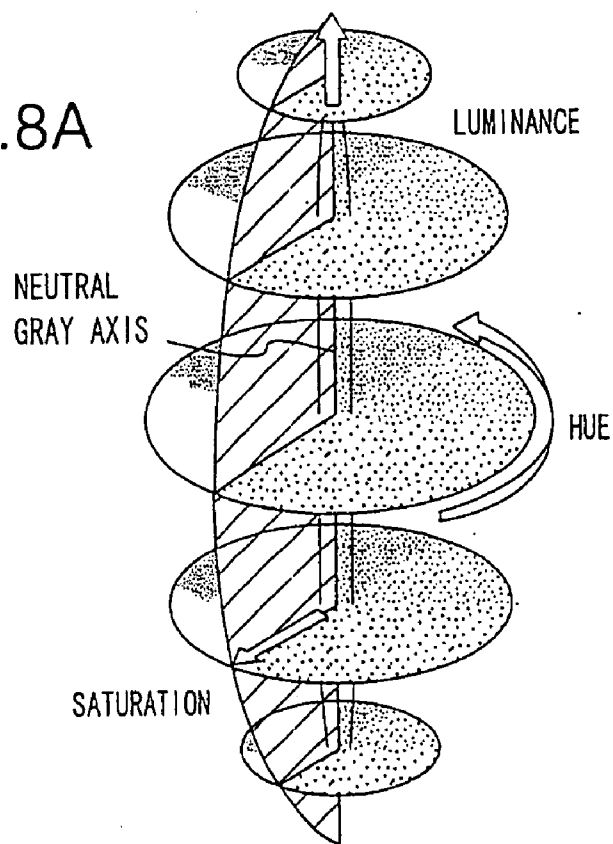
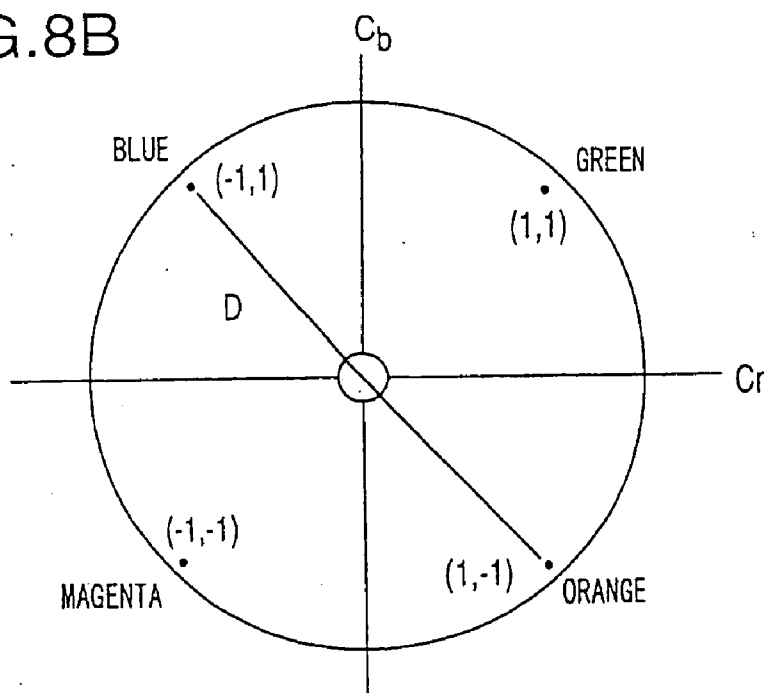


FIG.8B



SYSTEM AND METHOD FOR OBJECT TRACKING

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] The present application is a continuation and claims the priority benefit of U.S. patent application Ser. No. 10/928,778 entitled "System and Method for Object Tracking," filed Aug. 26, 2004, which is a continuation and claims the priority benefit of U.S. patent application Ser. No. 09/621,578 entitled "Method for Mapping an Object from a Two-Dimensional Camera Image to a Three-Dimensional Space for Controlling Action in a Game Program," filed Jul. 21, 2000, and now U.S. Pat. No. 6,795,068. The disclosure of this commonly owned application is incorporated herein by reference.

[0002] This application is related to U.S. patent application Ser. No. 10/927,918 entitled "Method for Color Transition Detection," filed Aug. 26, 2004 and now U.S. patent number 7,_____, which is a divisional and claims the priority benefit of U.S. patent application Ser. No. 09/621,578 entitled "Method for Mapping an Object from a Two-Dimensional Camera Image to a Three-Dimensional Space for Controlling Action in a Game Program," filed Jul. 21, 2000, and now U.S. Pat. No. 6,795,068. The disclosure of this commonly owned application is incorporated herein by reference.

BACKGROUND OF THE INVENTION

[0003] 1. Field of the Invention

[0004] The present invention relates to computer vision systems, and more particularly to a system in which an object is picked-up via an individual video camera, the camera image is analyzed to isolate the part of the image pertaining to the object, and the position and orientation of the object is mapped into a three-dimensional space. A three-dimensional description of the object is stored in memory and used for controlling action in a game program, such as rendering of a corresponding virtual object in a scene of a video display.

[0005] 2. Background of the Invention

[0006] Tracking of moving objects using digital video cameras and processing the video images for producing various displays has been known in the art. One such application, for producing an animated video version of a sporting event, has been disclosed by Segen, U.S. Pat. No. 6,072,504, the disclosure of which is incorporated in the present specification by reference. According to this system, the position of a tennis ball during play is tracked using a plurality of video cameras, and a set of equations relating the three-dimensional points in the court to two-dimensional points (i.e. pixels) of digital images within the field of view of the cameras are employed. Pixel positions of the ball resolved in a given digital image can be related to a specific three-dimensional position of the ball in play and, using triangulation from respective video images, a series of image frames are analyzed by a least-squares method, to fit the positions of the ball to trajectory equations describing unimpeded segments of motion of the ball.

[0007] As described in some detail by Segen, once a three-dimensional description of position and motion of an

object has been determined, various methods exist which are well known in the art for producing an animated representation thereof using a program which animates appropriate object movement in a video game environment.

[0008] Stated otherwise, Segen is concerned with determining the three-dimensional position of an object in motion from a plurality of two-dimensional video images captured at a point in time. Once the three-dimensional position of the "real" object is known, it is then possible to use this information to control a game program in any number of different ways which are generally known to game programmers.

[0009] However, the system of Segen relies on a plurality of video cameras for developing positional information about the object based on triangulation. Moreover, the detected object of Segen is a simple sphere which does not require information about the orientation (e.g. inclination) of the object in space. Thus, the system of Segen is not capable of reconstructing position and orientation of an object, whether moving or at rest, from a two-dimensional video image using a single video camera.

[0010] It is common for game programs to have virtual objects formed from a combination of three-dimensional geometric shapes, wherein during running of a game program, three-dimensional descriptions (positions and orientations) of the objects relative to each other are determined by control input parameters entered using an input device such as a joystick, game controller or other input device. The three-dimensional position and orientation of the virtual objects are then projected into a two-dimensional display (with background, lighting and shading, texture, and so forth) to create a three-dimensional perspective scene or rendition by means of the rendering processor functions of the game console.

[0011] As an example, there can be "virtual object" that forms a moving image in a game display corresponding to how one moves around the "real" object. To display the virtual object, the calculated three-dimensional information is used for fixing the position and orientation of the "virtual object" in a memory space of the game console, and then rendering of the image is performed by known projection processing to convert the three-dimensional information into a realistic perspective display.

[0012] However, in spite of the above knowledge and techniques, problems continue to hinder successful object tracking, and a particularly difficult problem is extracting precisely only those pixels of a video image which correspond unambiguously to an object of interest. For example, although movement of an object having one color against a solid background of another color, where the object and background colors vary distinctly from one another, can be accomplished with relative ease, tracking of objects, even if brightly colored, is not so easy in the case of multi-colored or non-static backgrounds. Changes in lighting also dramatically affect the apparent color of the object as seen by the video camera, and thus object tracking methods which rely on detecting a particular colored object are highly susceptible to error or require constant re-calibration as lighting conditions change. The typical home use environment for video game programs demands much greater flexibility and robustness than possible with conventional object tracking computer vision systems.

SUMMARY OF THE INVENTION

[0013] In one exemplary embodiment of the present invention, an object tracking system is provided. The exemplary tracking system comprises an input device configured to detect two-dimensional input pixel data from a prop device. The system also comprises a multiprocessor unit configured to calculate three-dimensional position and orientation data associated with the prop device from the two-dimensional input pixel data.

[0014] The present invention also discloses an exemplary method for tracking an object. Through this exemplary method, pixel data is received from an input device. Edges of an object are defined from the received pixel data and three-dimensional position and orientation data of the object are calculated, wherein the edges of the object are associated with the three-dimensional position and orientation data of the prop device.

[0015] A machine readable medium having embodied thereon a program being executable by a machine to perform a method for tracking an object is also disclosed herein. That tracking method, in accordance with the present exemplary embodiment, generally corresponds to the aforementioned tracking method.

BRIEF DESCRIPTION OF THE DRAWINGS

[0016] FIG. 1 is a block diagram showing an example of a configuration of a main part of a video game console which is adapted to receive an input from a digital video camera.

[0017] FIG. 2 is an illustration showing movement of a hand held prop, as an auxiliary input device, in front of a digital camera for causing corresponding action on a video display of a game program.

[0018] FIG. 3 is a block diagram showing the functional blocks required for tracking and discrimination of the prop as it is manipulated by the user.

[0019] FIG. 4A illustrates a prop device according to one aspect of the present invention.

[0020] FIG. 4B illustrates a process for mapping two-dimensional pixel data of a cylinder corresponding to the prop device shown in FIG. 4A to a three-dimensional space.

[0021] FIG. 5A illustrates a prop device according to another aspect of the present invention.

[0022] FIG. 5B illustrates a process for mapping two-dimensional pixel data of a combined sphere and cylinder corresponding to the prop device shown in FIG. 5A to a three-dimensional space.

[0023] FIG. 6A illustrates a prop device according to still another aspect of the present invention.

[0024] FIG. 6B illustrates a process for mapping two dimensional pixel data of stripes provided on a cylinder corresponding to the prop device shown in FIG. 6A to a three-dimensional space on the basis of color transitions at the stripes.

[0025] FIG. 7 illustrates a prop device having a helical stripe thereon, and provides a description of principles of another aspect of the present invention whereby a rotational component of the prop can be determined.

[0026] FIGS. 8A and 8B are graphs for describing a two-dimensional chrominance color space, for illustrating principles by which color transitions associated with colored stripes provided on a manipulated object are selected to maximize their detectability.

DETAILED DESCRIPTION

[0027] FIG. 1 is a block diagram of a configuration of a main part of a video game console 60 adapted for use with a manipulated object (prop) serving as an alternative input device.

[0028] The game console 60 constitutes a component of an overall entertainment system 110 according to the present invention which, as shown in FIG. 1 is equipped by a multiprocessor unit MPU 112 for control of the overall system 110, a main memory 114 which is used for various program operations and for storage of data, a vector calculation unit 116 for performing floating point vector calculations necessary for geometry processing, an image processor 120 for generating data based on controls from the MPU 112, and for outputting video signals to a monitor 80 (for example a CRT), a graphics interface (GIF) 112 for carrying out mediation and the like over a transmission bus between the MPU 112 or vector calculation unit 116 and the image processor 120, an input/output port 124 for facilitating reception and transmission of data to and from peripheral devices, an internal OSD functional ROM (OSDROM) 126 constituted by, for example, a flash memory, for performing control of a kernel or the like, and a real time clock 128 having calendar and timer functions.

[0029] The main memory 114, vector calculation unit 116, GIF 112, OSDROM 126, real time clock 128, and input/output port 124 are connected to the MPU 112 over a data BUS 130.

[0030] Further connected to the BUS 130 is an image processing unit 138 which is a processor for expanding compressed moving images and texture images, thereby developing the image data. For example, the image processing unit 138 can serve functions for decoding and development of bit streams according to the MPEG2 standard format, macroblock decoding, performing inverse discrete cosine transformations, color space conversion, vector quantization and the like.

[0031] A sound system is constituted by a sound processing unit SPU 171 for generating musical or other sound effects on the basis of instructions from the MPU 112, a sound buffer 173 into which waveform data may be recorded by the SPU 171, and a speaker 175 for outputting the musical or other sound effects generated by the SPU 171. It should be understood that the speaker 175 may be incorporated as part of the display device 80 or may be provided as a separate audio line-out connection attached to an external speaker 175.

[0032] A communications interface 140 is also provided, connected to the BUS 130, which is an interface having functions of input/output of digital data, and for input of digital contents according to the present invention. For example, through the communications interface 140, user input data may be transmitted to, and status data received from, a server terminal on a network. An input device 132 (also known as a controller) for input of data (e.g. key input

data or coordinate data) with respect to the entertainment system **110**, an optical disk device **136** for reproduction of the contents of an optical disk **70**, for example a CD-ROM or the like on which various programs and data (i.e. data concerning objects, texture data and the like), are connected to the input/output port **124**.

[0033] As a further extension or alternative to the input device, the present invention includes a digital video camera **190** which is connected to the input/output port **124**. The input/output port **124** may be embodied by one or more input interfaces, including serial and USB interfaces, wherein the digital video camera **190** may advantageously make use of the USB input or any other conventional interface appropriate for use with the camera **190**.

[0034] The above-mentioned image processor **120** includes a rendering engine **170**, a main interface **172**, an image memory **174** and a display control device **176** (e.g. a programmable CRT controller, or the like).

[0035] The rendering engine **170** executes operations for rendering of predetermined image data in the image memory, through the memory interface **172**, and in correspondence with rendering commands which are supplied from the MPU **112**.

[0036] A first BUS **178** is connected between the memory interface **172** and the rendering engine **170**, and a second BUS is connected between the memory interface **172** and the image memory **174**. First BUS **178** and second BUS **180**, respectively, have a bit width of, for example **128** bits, and the rendering engine **170** is capable of executing high speed rendering processing with respect to the image memory.

[0037] The rendering engine **170** has the capability of rendering, in real time, image data of 320×240 pixels or 640×480 pixels, conforming to, for example, NTSC or PAL standards, and more specifically, at a rate greater than ten to several tens of times per interval of from $\frac{1}{60}$ to $\frac{1}{30}$ of a second.

[0038] The image memory **174** employs a unified memory structure in which, for example, a texture rendering region and a display rendering region, can be set in a uniform area.

[0039] The display controller **176** is structured so as to write the texture data which has been retrieved from the optical disk **70** through the optical disk device **136**, or texture data which has been created on the main memory **114**, to the texture rendering region of the image memory **174**, via the memory interface **172**, and then to read out, via the memory interface **172**, image data which has been rendered in the display rendering region of the image memory **174**, outputting the same to the monitor **80** whereby it is displayed on a screen thereof.

[0040] There shall now be described, with reference to **FIG. 2**, an overall system configuration by which a user holding a prop object manipulates the object in front of a digital video camera, for causing an action to occur in a video game.

[0041] As shown in **FIG. 2**, the prop may comprise a stick-like object which is made up of a handle **303** which is typically black in color, and a brightly colored cylinder (i.e. having a saturated color) **301**. A user stands in front of the video camera **190**, which may comprise a USB webcam or a digital camcorder connected to an input/output port **124** of

a game console **60** such as the "Playstation 2" manufactured by Sony Computer Entertainment Inc. As the user moves the object in front of the camera **190**, the features of the object relating to the cylinder are picked up by the camera **190**, and processing (to be described later) is performed in order to isolate and discriminate a pixel group corresponding only to the cylinder. A three-dimensional description of the cylinder, including its position and orientation in three-dimensional space, is calculated, and this description is correspondingly stored in the main memory **114** of the game console **60**. Then, using rendering techniques known in the art, the three-dimensional description of the object is used to cause action in a game program which is displayed on the display screen of the monitor **80**. For example, a virtual object, shown as a torch for example, can be moved throughout the scene of the game, corresponding to the movements of the real object made by the user. As the user changes the position and orientation of the object by moving it, the three-dimensional description of the object in the memory **114**, and a corresponding rendering of the object in the rendering area of image memory **174**, are continuously updated so that the position and orientation of the virtual object, or torch, on the monitor **80** changes as well.

[0042] As noted above, the essential information which must be provided is a three-dimensional description of the object, which in the case of **FIG. 2** is a cylinder. However, the image which is picked up by the camera provides only two-dimensional pixel information about the object. Moreover, it is necessary to discriminate the pixels which relate only to the object itself before a three-dimensional description thereof can be calculated.

[0043] **FIG. 3** is a block diagram showing the functional blocks used to track and discriminate a pixel group corresponding to the prop as it is being manipulated by the user. It shall be understood that the functions depicted by the blocks are implemented by software which is executed by the MPU **112** in the game console **60**. Moreover, not all of the functions indicated by the blocks in **FIG. 3** are used for each embodiment. In particular, color transition localization is used only in the embodiment described in relation to **FIGS. 6A and 6B**, which shall be discussed below.

[0044] Initially the pixel data input from the camera is supplied to the game console **60** through the input/output port interface **124**, enabling the following processes to be performed thereon. First, as each pixel of the image is sampled, for example, on a raster basis, a color segmentation processing step **S201** is performed, whereby the color of each pixel is determined and the image is divided into various two-dimensional segments of different colors. Next, for certain embodiments, a color transition localization step **S203** is performed, whereby regions where segments of different colors adjoin are more specifically determined, thereby defining the locations of the image in which distinct color transitions occur. Then, a step for geometry processing **S205** is performed which, depending on the embodiment, comprises either an edge detection process or performing calculations for area statistics, to thereby define in algebraic or geometric terms the lines, curves and/or polygons corresponding to the edges of the object of interest. For example, in the case of the cylinder shown in **FIG. 2** the pixel area will comprise a generally rectangular shape corresponding to an orthogonal frontal view of the cylinder. From the algebraic or geometric description of the rectangle, it is possible to

define the center, width, length and two-dimensional orientation of the pixel group corresponding only to the object.

[0045] The three-dimensional position and orientation of the object are calculated in step S207, according to algorithms which are to be described in association with the subsequent descriptions of preferred embodiments of the present invention.

[0046] Lastly, the data of three-dimensional position and orientation also undergoes a processing step S209 for Kalman filtering to improve performance. Such processing is performed to estimate where the object is going to be at a point in time, and to reject spurious measurements that could not be possible, and therefore are considered to lie outside the true data set. Another reason for Kalman filtering is that the camera 190 produces images at 30 Hz, whereas the typical display runs at 60 Hz, so Kalman filtering fills the gaps in the data used for controlling action in the game program. Smoothing of discrete data via Kalman filtering is well known in the field of computer vision and hence will not be elaborated on further.

[0047] In FIG. 4A, a prop which is used according to the first embodiment shall be described, and in FIG. 4B a description is given which explains how three-dimensional information of the position and orientation of the prop of FIG. 4A is derived from a two-dimensional video image thereof.

[0048] As shown in FIG. 4A, the prop is a cylindrical body 301 having a single solid color attached to a handle 303 which is preferably black in color. In order to fully define the position and orientation of the object in a three-dimensional space, a position of a given point p, typically the center of the object, in the X-Y plane and a depth quantity Z (i.e. the position of the point p on the Z axis) must be determined, together with angular information of the object in at least two different planes, for example, an inclination θ of the object in the X-Y plane, and an inclination ϕ of the object in the Y-Z plane. The actual physical length and diameter of the cylinder 301, together with knowledge of the focal length of the camera, may be used for scaling, but are not essential for programming action in a game program since the virtual object shown on the display need not be of the same length and diameter, or even of the same shape, as the prop.

[0049] Referring now to FIG. 4B, this figure shows a two-dimensional pixel image 305 of the object produced by the video camera 190. A frontal orthogonal view of the cylindrical object 301 is picked up in the video image which appears as a generally rectangular pixel group 307, however, wherein the width of the pixel group can vary along the length l thereof as a result of the object being inclined in the phi ϕ direction or as a result of the distance overall of the prop from the camera. It will be understood that the inclination in the phi ϕ direction is not directly visible in the video image 305.

[0050] To determine the length, center point, etc. of the pixel group 307 in accordance with the geometry processing step S205 discussed above, known area statistics calculations are used. Area statistics include the area, centroid, moment about the X-axis, moment about the Y-axis, principal moments, and the angle of principal moments, which typically are used for calculating moments of inertia of objects about a certain axis. For example, to determine the

moments about the X and Y axes, respectively, if each pixel making up the pixel group is considered to correspond to a particle of a given uniform mass m in making up a thin homogeneous sheet or lamina, then the moments about x and y axes of a system of n such particles (or pixels) located in a coordinate plane are defined as follows:

$$M_x = \sum_{k=1}^n \cdot y_k \cdot m_k \tag{1}$$

$$M_y = \sum_{k=1}^n \cdot x_k \cdot m_k \tag{2}$$

[0051] The center of mass of this system is located at the point (x, y) given by

$$x = \frac{M_y}{m}, \quad y = \frac{M_x}{m} \tag{3}$$

[0052] Further, assuming the lamina is of a shape having a geometric center, such as the rectangle in the case of FIG. 4B or a circle in the case of FIG. 5B (to be discussed later), the center of mass of such a lamina corresponds to the geometric center. More generally, if one knows the area statistics of the pixel region and, for example, that the two-dimensional shape is a rectangle, one can directly calculate its width, height and orientation. Similar calculations are possible with circular shapes to determine the center point and radius, for example. Representative calculations for cases of rectangles and circles can be found in standard college-level calculus or physics texts.

[0053] Because the image 305 is already taken to be in the X-Y plane, the X-Y position of the center point p can be derived directly from the image. Also, the theta θ quantity is taken directly from the image simply by knowing any line l, determined in accordance with the geometry processing step S205 described above, which runs along the longitudinal axis of the pixel group 307 corresponding to the cylinder 301. Typically, a longitudinal line l passing through the center point p is used for this purpose.

[0054] Determination of the phi ϕ quantity requires some additional knowledge about the pixel width w in at least two different locations W1 and W2 wherein the ratio of the width quantities $w_1:w_2$ provides a value which can be used for determining ϕ . More specifically, if the cylinder 301 is inclined so that the top end thereof is closer to the camera 190 than the lower end of the cylinder, then, since the lower end of the cylinder is at a greater distance from the camera 190, the pixel width quantity W_2 of the image will have a greater value than the pixel width quantity w_1 , and vice versa. The ratio $w_1:w_2$ is proportional to the inclination ϕ of the cylinder 301 in the Y-Z plane, and therefore the phi ϕ quantity ϕ can be determined from this ratio. Typically, for better accuracy, a plurality of equidistant measurements of pixel widths between ends of the pixel group 307 are taken, and averaging is performed to determine the ratio $w_1:w_2$.

[0055] Determination of the depth quantity Z can be done in different ways. However, it is important to recognize that

the size and number of pixels making up the two-dimensional pixel group 307 are affected both by the inclination of the object in the ϕ direction as well as by the actual distance of the physical object from the video camera 190. More specifically, as the object inclines in the ϕ direction, the apparent length of the object as seen by the video camera tends to shorten, so that the length l of the pixel group shortens as well. However, at the same time, as the object moves farther away from the camera along the Z-axis, the apparent size of the object overall, including its length l , also becomes smaller. Therefore, it is insufficient simply to look at the length l alone as an indicator of how far away from the camera the object is. Stated otherwise, the depth quantity Z must be determined as a function of both l and ϕ .

[0056] However, if the phi quantity ϕ has already been determined and is known, a phi-weighted value of l , which we may call $l\phi$, can be determined, and the pixel length of $l\phi$ in the image, which changes as the object is moved closer or farther from the camera while assuming that ϕ stays constant, then can be used to determine the depth quantity Z since $l\phi$, will be proportional to Z .

[0057] Another method for determining depth Z is to count the total number of pixels in the pixel group 307 corresponding to the object. As the object gets closer to or farther away from the camera, the number of pixels making up the pixel group 307 increases or decreases respectively, in proportion to the depth quantity Z . However, again, the number of pixels in the pixel group 307 is also affected by the inclination in the phi ϕ direction, so the number of pixels N must first be weighted by phi ϕ to result in a weighted quantity $N\phi$ which is used for determining the depth quantity Z based on a proportional relationship between $N\phi$ and Z .

[0058] Yet another advantageous method for determining the depth quantity Z is to use the average width w_{avg} of the rectangle, which is calculated as the sum of a given number of width measurements of the rectangle divided by the number of width measurements taken. It should be clear that the average width of the pixel group is affected only by Z and not by the phi-inclination of the cylinder. It is also possible to determine phi ϕ from the ratio of the total length of the pixel group to the average width (i.e. $l: w_{avg}$), and moreover, wherein the sign of the phi-inclination can be determined based on whether w_1 is greater or less than w_2 .

[0059] In FIG. 5A, a prop which is used according to another embodiment shall be described, and in FIG. 5B a description is given which explains how three-dimensional information of the position and orientation of the prop of FIG. 5A is derived from a two-dimensional video image thereof.

[0060] The prop according to the second embodiment, similar to the first embodiment shown in FIG. 4A, comprises a cylindrical stick-shaped object, however in this case, a spherical object 309 of a different color is rigidly affixed to one end of the cylinder 301. In addition, although not shown, a distal end of the cylinder may be provided which protrudes just slightly and is visible from an upper end of the sphere 309. As shall be explained below, the sphere 309 provides a simplified means for determining the depth quantity Z and the inclination of the object in the phi ϕ direction, which does not require measurement of relative widths of the cylinder 301, and which does not require any weighting of the length quantity by phi ϕ in order to determine the depth quantity Z .

[0061] As shown in FIG. 5B, a pixel group 311 corresponding to the sphere 309 in the image appears as a two-dimensional circle. According to this embodiment, a radius R and center point p_s of the circle are determined according to area statistics calculations which have already been discussed above. In this case, further, the total number of pixels making up the pixel group 311 of the circle can be counted for giving a pixel area of the circle. It will be appreciated that the circular pixel area will increase as the spherical object 309 comes closer to the camera 190 and vice versa, and therefore, since the total number of pixels in the pixel group 311 making up the circle is proportional to the depth quantity Z , the value for Z can be determined thereby.

[0062] It should also be realized that, unlike the cylinder in the previous embodiment, the shape and size of the circular pixel group are not influenced as a result of the phi ϕ angle of inclination. More specifically, even if the object overall is tilted in the phi direction, the sphere 309 and the pixel group 311 will retain their general shape and, unlike the length of the cylinder 301, will not become foreshortened as a result of such tilting. Therefore, an advantage is obtained in that the total number of pixels of the pixel group making up the circle in the image can always be related proportionately to the depth quantity Z and, for determining Z , phi-weighting as in the previous embodiment is not required.

[0063] Determination of inclination of the object in the theta θ direction is done directly from the image, just as in the previous embodiment, by determining the angle theta θ between a center longitudinal line of the pixel group 307 corresponding to the cylinder 301 of the Y-axis.

[0064] Determining the angle of inclination in the phi ϕ direction is handled somewhat differently than the previous embodiment. More specifically, such a quantity can be determined by knowledge of the depth quantity Z , determined as described above, and by the length l between the center point of the circle 311 and the center point of the pixel group 307 which corresponds to the cylinder 301. For any known and fixed depth quantity Z , the length l (as viewed from the perspective of the camera) becomes shorter as the object is tilted in the phi ϕ direction. Therefore, if the Z quantity is known, it is possible to determine, simply from the length l , the degree of inclination in the phi ϕ direction, and it is not necessary to calculate a relative width quantity of ratio of widths, as in the embodiment shown by FIGS. 4A and 4B.

[0065] FIG. 6A illustrates a prop device according to still another aspect of the present invention.

[0066] As in the embodiment shown in FIG. 6A, the prop itself comprises a generally cylindrical body 301. In addition, three stripes S_1 , S_2 and S_3 having a different color than the cylinder itself are provided on the cylindrical body. Preferably, the stripes S_1 , S_2 and S_3 are each equal in width and are spaced equidistant from each other, at either end of the cylinder 301 and in the center thereof.

[0067] According to this embodiment, a pixel group making up the cylinder is extracted from the image to provide a two-dimensional line along which to look for color transitions. To determine the quantities Z , θ and ϕ , positions are determined at which color transitions along any line l in the longitudinal direction of the cylinder 301 occur.

[0068] More specifically, as shown in **FIG. 6B**, a group made up of only those pixels corresponding to a line *l* along the longitudinal direction of the cylinder body **301**, as viewed by the camera, needs to be sampled in order to determine where along the line *l* distinct color transitions occur. In particular, for detecting such color transitions, the chrominance values Cr and Cb which are output as part of the YCrCb signals from the video camera **190** are detected. For reasons which shall be explained below in connection with the criteria for selecting the stripe colors, it is preferable to use a combined chrominance value *D* made up of a Pythagorean distance of the combined chrominance signals Cr and Cb for each color of the cylinder **301** and stripes *S*₁, *S*₂ and *S*₃, respectively, thereby defining a separation in the two-dimensional chrominance color space used by the video camera **190**, according to the following formula (1):

$$D = \sqrt{(\Delta Cr)^2 + (\Delta Cb)^2} \quad (4)$$

[0069] By selecting colors which maximize the value of *D* (to be explained in more detail later), it is possible to select a threshold *D*_t at which only color transitions above a certain separation where *D* > *D*_t are considered to correspond to the color transitions of the stripes *S*₁, *S*₂ and *S*₃. Accordingly, the pixels along the line of the cylinder are filtered, using such a threshold, in order to find the large color transitions corresponding to the stripes *S*₁, *S*₂ and *S*₃.

[0070] As shown in **FIG. 6B**, at positions along the line *l* where color transitions occur, for each stripe two spikes corresponding to positions where color transitions appear can be detected, and the center point between these spikes is taken to be the position of the stripes. Once the positions of the stripes are fixed, it is then a matter of course to determine the lengths *l*₁ and *l*₂ between the stripes, wherein the overall length of the cylinder is determined by the sum of *l*₁ and *l*₂.

[0071] It shall next be explained how knowledge of *l*₁ and *l*₂ provides sufficient information for determining the quantities of *Z*, θ and Φ , necessary for describing the position and orientation of the object in three dimensions.

[0072] First, since the line *l* defined by the pixels running along the length of the cylinder has already been determined, and since the camera is assumed to face normally to the X-Y plane, the angle θ is taken directly as the angle between the longitudinal line of the cylinder and the Y axis, basically in the same manner as the preceding embodiments.

[0073] For determining the angle of inclination in the phi ϕ direction, the ratio of the lengths *l*₁:*l*₂ is used. For example, in the case (as shown) in which the cylinder is inclined in the ϕ direction toward the camera **190**, with the upper end of the cylinder being closer to the camera than the lower end, the length *l*₁ will appear longer to the camera **190** (since it is closer) than the length *l*₂. It will also be appreciated that, although the apparent lengths *l*₁ and *l*₂ will also be affected by the overall distance *Z* of the object from the camera **190**, the ratio of these lengths *l*₁:*l*₂ will not change and therefore this ratio provides a constant indication of the inclination of the cylinder **301** in the phi ϕ direction.

[0074] For determining the depth quantity *Z*, a procedure similar to that of the first embodiment is employed, wherein a phi-weighted quantity *l* _{ϕ} of the total length *l* (*l* = *l*₁ + *l*₂) is determined for giving *Z*. More specifically, the influence of the inclination angle ϕ on the total apparent length *l* of the object is first determined, and then the total length, properly

weighted by the influence of ϕ , is proportional to the distance (or depth quantity) *Z* of the object from the camera **190**.

[0075] Stated more simply, ϕ is determined from the ratio of *l*₁ and *l*₂, and once phi ϕ is known, the total depth quantity *Z* can be determined from the sum of *l*₁ and *l*₂.

[0076] There shall now be described, in connection with **FIG. 7**, a method for determining a rotational component of the prop. This method may be applied in conjunction with any of the embodiments which have discussed above, by further equipping the prop device with a helical stripe *S*_H thereon.

[0077] Each of the tracking methods described above can be used to obtain five of the six degrees of freedom of the objects. The only one missing is the rotation of the cylinder about its axis. Information about the rotation of the cylinder would seem difficult to obtain because cylinders are symmetric in rotation about this axis. The approach taken by the present invention to obtain this rotational component is to add a helical stripe *S*_H that goes around the cylinder **301** exactly once. As the cylinder **301** is rotated, the height of the stripe *S*_H will correspond to the rotation angle.

[0078] More specifically, as shown in **FIG. 7**, the cylinder **301** (or the cylinder-part of the prop in the case of **FIGS. 5A and 5B**) includes the single helical strip *S*_H thereon which goes around the object only once. Information pertaining to the helical stripe is extracted, either from the entire pixel group **313** which makes up the helical stripe or by using the color transitions corresponding to the helical stripe *S*_H, in order to determine, using the geometry processing discussed above, a helix *H* which best fits to the stripe *S*_H.

[0079] In addition to the helix *H*, a center line *l* of the pixel group corresponding to the cylinder is determined as described previously. Also the overall length *l* of the pixel group is determined.

[0080] For obtaining a degree of rotation of the cylinder, various heights *h* (only *h*₁ and *h*₂ are shown for simplicity) each of which define the distance between one end of the cylinder and the point *p* where the center line intersects the helix are determined.

[0081] As shown on the right-hand side of **FIG. 7**, the camera **190** only sees one side (or orthogonal projection) of the cylinder **301** at a time. Accordingly, the helix *H* determined from the extracted region of the camera image determines the degree of revolution of the cylinder **301**. More specifically, as shown, assuming no rotation (i.e. a rotational component of 0 degrees), a center line extending from one end to a point on the helix will have a first height *h*₁, whereas if the object is rotated by 45 degrees, the height of the center line *l* between the lower end to the point where it intersects the helix *H* will have a shorter height *h*₂. The condition shown by the far right-hand side of **FIG. 7**, at a rotation of 90 degrees, represents to a unique case in which the center line will intersect the helix at two points. Hence, by calculating the heights of the center line *l*, a component of rotation of the cylinder **301** (or any other object affixed to the cylinder and rotated thereby) can be determined.

[0082] The specific quantity used for determining rotation is the ratio of the detected height between the lower end and the point on the helix to the total length *l* of the pixel group.

This ratio gives a number from 0 to k (where $k=h_{\max}/1$), which maps directly to a range of from 0 to 360 degrees. Thus, additional information with respect to the object and orientation of the cylinder **301** in three-dimensional space can be provided. Such information can be used to control the rotation of a virtual object, for example, when displayed in a game program.

[0083] Next, with respect to **FIGS. 8A and 8B**, a process for selection of colors for the stripes, according to the embodiments of **FIGS. 6A and 6B** shall now be described. More specifically, **FIG. 8A** shows a diagram of a color space defined by luminance and radial coordinates of hue and saturation. Luminance is the brightness or intensity of the color, hue is the shift in the dominant wavelength of a spectral distribution, and saturation is the concentration of a spectral distribution at one wavelength.

[0084] By contrast, **FIG. 8B** shows a two-dimensional chrominance color space corresponding to the Cr and Cb chrominance output signals of the video camera. It is well understood in the art that video cameras output signals for controlling the color of each pixel making up a video image. As shown by the color wheel diagram of **FIG. 8A**, color can be defined using radial coordinates corresponding respectively to hue and saturation. However, as it is needlessly complex for computerized image processing to use radial coordinates, another more useful standard for defining color is the YCrCb color definition, which is the most common representation of color used in the video world. YCrCb represents each color by a single luma component (Y) and two components of chrominance Cr and Cb. Y may be loosely related to brightness of luminance whereas Cr and Cb make up a quantities loosely related to hue. These components are defined more rigorously in ITU-R BT.601-4 (Studio encoding parameters of digital television for standard 4:3 and wide-screen 16:9 aspect ratios) published by the International Telecommunication Union. Thus, the Cr and Cb chrominance signals for each pixel are defined by Cartesian coordinates which also can be used to determine a location within the color wheel corresponding to a certain hue and saturation.

[0085] According to the present invention, the color of the stripes S_1 , S_2 and S_3 and the color of the cylinder **301** are chosen in such a way as to maximize stripe detectability for the video camera. Color-based tracking is notorious for its difficulties due to changes in lighting, which cause the apparent color to vary. As a result, if one is attempting to detect a certain color of blue corresponding to an object, for example, under certain lighting conditions it is possible for the color of blue, as perceived by the camera, to vary to such a degree that accurate detection of the object is made difficult. In the present invention, by looking for color transitions instead of absolute colors, a more robust tracking solution can be attained. For example, in the embodiment of **FIGS. 6A and 6B**, if the cylinder **301** is blue and the stripes S_1 , S_2 and S_3 are orange, if lighting conditions change, then the apparent colors will also change. However, the transition between these colors, as shown in **FIG. 6B**, will still be very evident.

[0086] As discussed above, video cameras capture data using the two-dimensional chrominance color space shown in **FIG. 8B**. By choosing colors for the object and stripes,

respectively, which have a maximal separation D in this space, it is possible to significantly enhance the detectability of the color transitions.

[0087] More specifically, as shown in **FIG. 8B**, highly saturated colors of blue and orange are located at substantially diametrically opposed sides of the color wheel and are separated by a large distance D in the color space. The actual distance D can be calculated as the hypotenuse of a triangle having sides defined by ΔCr (i.e. the difference in the Cr chrominance signal values for the two colors of blue and orange) and ΔCb (i.e. the difference in the Cb chrominance signal values for the same two colors), and hence the actual distance D is the square root of $(\Delta Cr)^2 + (\Delta Cb)^2$, as already discussed above in equation (4).

[0088] Although blue and orange have been described as an example, it will be appreciated that any other color pairs, for example green and magenta, which also possess a large separation in the chrominance color space may be used. In other words, the method provides a general criteria whereby colors may be selected using their chrominance signals Cr and Cb in such a manner to maximize their separation in the color space.

[0089] More specifically, a generally applicable method for the selection of colors, as well as for calculating distance between any two colors, is performed in such a way that the distance between two colors is calculated as a distance projected onto a certain diameter-spoke of the color wheel. First, a given diameter-spoke on the color wheel is selected having a certain angle of orientation θ . By choosing the angle of orientation of the selected diameter on the color wheel, it is possible to select the color transitions one wants to detect. For example, if green is (1, 1) and magenta is (-1, -1), the diameter of the spoke should be set at an orientation θ of 45 degrees. Then the color separation distance is calculated simply by projecting the colors onto the 45 degree line. In this manner, for the case of green and magenta, the computed distance is exactly the same as the Pythagorean distance D discussed above, however with a diameter-line orientation of 45 degrees, the distance between blue and orange is zero, because they both project to the origin. This tells us that, for a selected diameter line of 45 degrees, green and magenta are the optimal colors for detection, since they possess the maximum separation in the color space for this diameter.

[0090] Thus, for any given diameter angle of θ , which can be chosen from 0 to 180 degrees, the separation between two colors (Cr_1, Cb_1) and (Cr_2, Cb_2) may be calculated according to equation (5) as follows:

$$D = |Cr_1 \cdot \cos \theta + Cb_1 \cdot \sin \theta| - |Cr_2 \cdot \cos \theta + Cb_2 \cdot \sin \theta| \quad (5)$$

[0091] The distance calculation shown by equation (5) can therefore also be used for setting the threshold D_t based on a predetermined orientation defined by the angle θ . For example, if the color transitions for the object were in fact green and magenta, the general distance calculation above can be used for threshold setting, while fixing the angle θ of this equation at 45 degrees.

[0092] Herein have been described several methods for determining the position and orientation of a real object manipulated in front of a video camera, by mapping the two-dimensional image information of the object captured by the camera to a three-dimensional space, wherein a three

dimensional description including position and orientation of the object may be used to control action in a game program.

[0093] Although one clear example of controlling a game program is to have a “virtual object” that forms a moving image in a game display corresponding to how the “real” object is moved or positioned, it will be appreciated that the three-dimensional information can be used to control game programs in any number of different ways foreseeable to persons skilled in the art. For example, a “theremin” like musical effect can be achieved wherein changes in the position and orientation of the manipulated object could be used to influence volume, tone, pitch, rhythm and so forth of sounds produced by the sound processor. Such a musical or rhythmic sound effect can be provided in combination with visual effects displayed on the screen of the game console, for enhancing the experience perceived by the game player.

[0094] It shall be understood that other modifications will be apparent and can be easily made by persons skilled in the art without departing from the scope and spirit of the present invention. Accordingly, the following claims shall not be limited by the descriptions or illustrations set forth herein, but shall be construed to cover with reasonable breadth all features which may be envisioned as equivalents by those skilled in the art.

What is claimed is:

- 1. An object tracking system comprising:
 an input device configured to detect two-dimensional input pixel data from a prop device; and
 a multiprocessor unit configured to calculate three-dimensional position and orientation data associated with the prop device from the two-dimensional input pixel data.
- 2. The object tracking system of claim 1, wherein the multiprocessor unit comprises a memory configured to store the three-dimensional position and orientation data associated with the prop device.
- 3. The object tracking system of claim 1, wherein the multiprocessor unit comprises an image processor configured to execute operations for rendering the three-dimensional position and orientation data associated with the prop device.
- 4. The object tracking system of claim 1, further comprising a monitor for displaying action in a game program caused by rendering the three-dimensional position and orientation data associated with the prop device.
- 5. The object tracking system of claim 1, wherein the prop device comprises a saturated color.
- 6. The object tracking system of claim 1, wherein the input device is a camera.
- 7. The object tracking system of claim 2, wherein the memory further comprises a display rendering region.
- 8. The object tracking system of claim 2, wherein the memory further comprises a texture rendering region.
- 9. The object tracking system of claim 1, wherein the multiprocessor unit is further configured to:

determine the color of each pixel in the two-dimensional input pixel data; and

define edges of an object by dividing an image comprising the input pixel-data into two-dimensional segments of color, wherein the defined edges are associated with the three-dimensional position and orientation data of the prop device.

10. The object tracking system of claim 9, wherein the multiprocessor unit is further configured to localize color transitions whereby distinct color transitions are defined prior to defining the edges of the object in the image.

11. The object tracking system of claim 1 further comprising a filter, wherein the filter is configured to filter the three-dimensional position and orientation data.

12. The object tracking system of claim 11, wherein the filter comprises a Kalman filter.

13. The object tracking system of claim 9, wherein the multiprocessor unit is further configured to employ an edge detection process to define the edges of the object.

14. The object tracking system of claim 9, wherein the multiprocessor unit is further configured to employ area statistics calculations to define the edges of the object.

15. The object tracking system of claim 9, wherein the definition of an edge of the object is algebraic.

16. The object tracking system of claim 9, wherein the definition of an edge of the object is geometric.

17. A method for tracking an object, comprising:

receiving pixel data from an input device;

defining edges of an object from the received pixel data; and

calculating three-dimensional position and orientation data of the object, wherein the defined edges are associated with the three-dimensional position and orientation data of the object.

18. The method of claim 17, further comprising localizing color transitions whereby distinct color transitions are defined prior to defining the edges of the object.

19. The method of claim 17, further comprising the application of a Kalman filter to the three-dimensional position and orientation data.

20. The method of claim 17, wherein defining the edges of the object comprises employing an edge detection process.

21. The method of claim 17, wherein defining the edges of the object comprises employing area statistics calculations.

22. A machine readable medium having embodied thereon a program being executable by a machine to perform a method for tracking an object, the method comprising:

receiving pixel data from an input device;

defining edges of an object from the received pixel data; and

calculating three-dimensional position and orientation data of the object, wherein the defined edges are associated with the three-dimensional position and orientation data of the object.

* * * * *