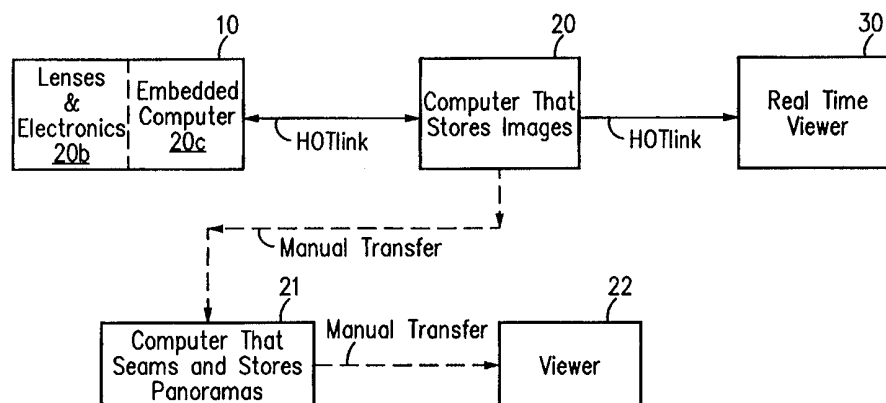




## INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification <sup>6</sup> :  <b>G03B</b>	<b>A2</b>	(11) International Publication Number: <b>WO 99/59026</b>  (43) International Publication Date: 18 November 1999 (18.11.99)
(21) International Application Number: PCT/US99/10403 (22) International Filing Date: 12 May 1999 (12.05.99) (30) Priority Data: 60/085,319                      13 May 1998 (13.05.98)                      US (71) Applicant: INFINITE PICTURES INC. [US/US]; Suite 1, 33 N.W. First Avenue, Portland, OR 97209 (US). (72) Inventors: GILBERT, Scott; 4309 North Rio Cancion Drive #183, Tucson, AZ 85719 (US). KAIMAN, David; 1694 North Island Cove Lane, Portland, OR 97219 (US). PARK, Michael, C.; 7000 S.W. Vermont Court #102, Portland, OR 97223 (US). RIPLEY, David, C.; 1546 N.W. Benfield Drive, Portland, OR 97229 (US). (74) Agent: GALBI, Elmer; 13314 Vermeer Drive, Lake Oswego, OR 97035 (US).		(81) Designated States: AU, BR, CA, CN, DE, ES, FI, IL, IN, JP, KP, KR, MX, NO, NZ, PT, RU, SE, TR, VN, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, 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), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).  <b>Published</b> <i>Without international search report and to be republished upon receipt of that report.</i>

(54) Title: PANORAMIC MOVIES WHICH SIMULATE MOVEMENT THROUGH MULTIDIMENSIONAL SPACE



## (57) Abstract

Movement through multidimensional space is simulated using a series of panoramic images which are projected or displayed in sequence. The user's direction of view, that is the selected view window, is maintained as the series of images is projected or displayed. Motion in directions other than forward or reverse is simulated by utilizing "branch" points in the sequence. Each path from a branch point simulates motion in a different direction. Branch points are generally indicated to a viewer by visual indicators called "hot spots"; however, branch points may also be hidden and activated in response to the viewer's selected direction of view. If a branch point is indicated by a visual indicator, a user can select motion in a desired direction by clicking on a "hot spot". In order to conserve storage space, the image representing each panorama can be stored in a compressed format. Only the portion of the panorama necessary to create a "view window" that is, the portion of the image displayed in response to the user's direction of view, is decompressed at view time. An index methodology is used to store the panoramic images. Use of the indexing methodology allows the images to be retrieved in both the forward and backward direction to simulate movement in either direction. Sound is provided in a special format, so that special effects can be provided based on the user's point of view and dependent upon the direction of motion selected by the user.

**FOR THE PURPOSES OF INFORMATION ONLY**

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

**PANORAMIC MOVIES WHICH SIMULATE MOVEMENT  
THROUGH MULTIDIMENSIONAL SPACE**

**Field of the Invention:**

The present invention relates to photography, digital image processing and to computer graphics. More particularly the present invention relates to a method and system for providing a viewer with a multidimensional view which simulates movement through space or time.

**Copyright Notice:**

A portion of the disclosure of this patent document contains material which is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent document; however, otherwise the copyright owner reserves all rights whatsoever.

**Background of the invention:**

Motion is usually simulated by means of single view movies. Single view movies consist of a series of single photographs which are sequentially projected on a screen. At any time, a single photograph is entirely projected on the screen. Some movie theaters have screens which partially surround the viewer, so that a viewer can turn and look at a different portion of the screen as the movie progresses. However, irrespective of where the viewer's attention is focused, in fact the entire image is being projected on the screen. With some equipment single view movies can be stopped and reversed; however, again at a particular time a selected frame is entirely displayed. In summary, traditional movies do not allow a viewer to control the portion of the image which is projected on the screen. Stated differently, with traditional movies a viewer can not control the "view window" through which each image is viewed.

It should be noted that as used herein the term "screen" refers to either a traditional screen onto which an image is projected or an electronic display which projects or displays an image in such a manner that the image can be seen by a viewer.

The technology for producing panoramic images and photographs is well known. Panoramic images are images which represent the visual surroundings from a single

1 location (or view point) of a particular 3D environment. Panoramic images can be  
2 photographic, computer-generated (CG), or a composite of photo and CG imagery.  
3 Equipment is available which seams together a series of two dimensional  
4 conventional images to form a panoramic image. Panoramic images may consist of  
5 any field of view such as a full sphere full cylinder, semi-sphere, etc. however, full  
6 sphere views are generally preferred. Panoramic images may be in any of the  
7 known projection formats, such as equi-rectangular, Mercator, Peters, fisheye, cube,  
8 or hemicube, etc. If the field of view is wide enough to warrant, perspective  
9 correction may be applied to the portion of a panoramic image displayed in order to  
10 remove projection distortions from the given view of the user. Computer programs  
11 and systems which allow one to view a selected portion of, to pan, to rotate, etc a  
12 panoramic image or photograph in response to the movement of a computer mouse,  
13 joystick, keyboard, etc. are commercially available.

14  
15 Panoramic image (or photographic image) viewing systems are available which  
16 provide a number of panoramas (for example views of adjacent rooms in a building)  
17 and which allow a user who is viewing one of the rooms to "click" on a door and to  
18 thus bring up the panorama of the next room, thereby, in some sense simulating  
19 movement into the next room. However, each of the panoramic views in such  
20 systems is static and some explicit action on the part of the viewer is required to  
21 move on to a different panorama.

22  
23 U.S. Patents 5,023,925 and 5,703,604 describe, and Dodeca L.L.C. located in  
24 Portland Oregon commercially markets, a system for capturing images using a multi  
25 lens camera, for seaming the images into panoramas, and for viewing selected  
26 portions of the panoramic images.

27  
28 Other panoramic image or photographic viewing systems are available which initiate  
29 a conventional single view motion picture when a user "clicks" on a spot such as a  
30 door in a first panoramic view. The single view movie simulates movement into a  
31 different location and at the end of the movie, the user is presented with a second  
32 panoramic view.

33  
34 A company named "Warp" located in the Kingdom of Tonga has demonstrated a  
35 system wherein a video sequence is captured using a video camera with a fisheye

1 lens which is pointed in the vertical or "up" direction (see, VTV Software  
2 Development Kit Reference Manual 2.01 Win95 1996). This approach provides a  
3 hemispheric movie in which the user may "pan around" while the movie is playing. In  
4 the system demonstrated by Warp, the user views the movie "in sequence", meaning  
5 that each frame of the movie is played according to the temporal sequence in which  
6 the hemispheric movie was captured. The system demonstrated by Warp was  
7 limited to sub-spherical panorama dimensions and the camera was located at a fixed  
8 position.

9  
10 Realistic simulation of movement from one location to another can be provided by  
11 three dimensional computer modeling systems, such as those used in some flight  
12 simulators. However, such systems are very computationally intensive

13  
14 **Summary of the present invention:**

15 The present invention simulates movement through multi-dimensional space using a  
16 series of panoramic images which are projected or displayed in sequence. The  
17 user's direction of view, that is the selected view window, is maintained as the series  
18 of images is projected or displayed. Motion in directions other than forward or  
19 reverse is simulated by utilizing "branch" points in the sequence. Each path from a  
20 branch point can simulate motion in a different direction. Branch points are generally  
21 indicated to a viewer by visual indicators called "hot spots"; however, branch points  
22 may also be hidden and activated in response to the viewer's selected direction of  
23 view. If a branch point is indicated by a visual indicator, a user can select motion in a  
24 desired direction by "clicking" on a "hot spot".

25  
26 In order to conserve storage space, the image representing each panorama can be  
27 stored in a compressed format. If the images are stored in compressed format, in  
28 order to conserve time and processing power, when an image is displayed, only the  
29 portion of the panorama necessary to create a "view window" that is, the portion of  
30 the image displayed in response to the user's direction of view, is decompressed.  
31 Furthermore, the images are stored in a format that does not utilize inter-image  
32 compression (such as that used by the MPEG format). Since the images are stored  
33 in a format that does not utilize inter-image compression, it is possible to simulate  
34 motion in both the forward and backward direction without operating on a series of  
35 decompressed images.

1  
2 An index methodology is used to store the panoramic images. Use of the indexing  
3 methodology allows the images to be retrieved in both the forward and backward  
4 direction to simulate movement in either direction.

5  
6 Sound is provided in a special format, so that special effects can be provided based  
7 on the user's point of view and dependent upon the direction of motion selected by  
8 the user.

9  
10 **Brief Description of the Figures:**

11 Figure 1 illustrates a key frame (that is, panoramic image) with a view window and  
12 associated sound tracks.

13  
14 Figure 2 is a block diagram showing the major components in the preferred  
15 embodiment.

16  
17 Figures 3A to 3D shows the sequence of operations performed by the various  
18 components in the system shown in Figure 2.

19  
20 Figure 4A illustrates a sequence of frames which constitute a panoramic movie.

21  
22 Figure 4B illustrates the sound track associated with the frames of a panoramic  
23 movie.

24  
25 Figure 5A is a perspective view of the multi lens hand held unit that captures a series  
26 of panoramic images.

27  
28 Figure 5B is top view of the multi lens hand held unit shown in Figure 5A.

29  
30 Figure 6 is a block diagram of the electronic components in the hand held unit shown  
31 in Figures 5A and 5B.

32  
33 Figure. 7 is a diagram of a file containing a pan movie. Figure 7 shows a series of  
34 panoramas stored as a series of compressed key-frames and a file index for  
35 sequencing playback of the key-frames.

Figure 8 is a block diagram of a program for inserting hot spots in a pan movie.

Figure 9A is a block diagram of a system for playback of a 3-D panoramic movie according to the invention.

Figure 9B is a block diagram of a real time viewing unit.

Figure 10 is a flowchart of the program for viewing a 3-D movie containing a sequence of panoramas according to the invention.

Figure 11 is a diagram illustrating the audio information associated with each key frame.

Figure 12A, 12B and 12C are a spatial sequence of perspectively correct views illustrating movement past a billboard displaying an advertisement which has been superimposed into a scene as a hot spot.

**Description of Appendices:**

Appendix A is a print out of computer code for retrieving images and correcting the perspective of images in a pan movie.

Appendix B is a sample of link control file for a pan movie.

Appendix C is a print out of computer pseudocode for linking sequences of images to form a pan movie.

**Description of a preferred embodiment:**

In order to simulate movement through multi-dimensional space, one must first capture a series of panoramic images, the panoramic images must be stored as frames and then the appropriate view window from selected frames must be displayed in an appropriate sequence.

A panoramic image provides data concerning what is visible in any direction from a particular point in space. At any particular time a viewer or user can only look in one direction. The direction or point of view of a viewer or user determines the "view window", that is, the part of a panoramic image which is projected on a screen at a

1 particular time. Figure 1 shows a key frame (i.e. a panoramic image) or a panorama  
2 3a. Panorama 3a has a view window 3b which corresponds to a portion of panorama  
3 3a. Panorama 3a also has associated therewith a number of sound tracks 3c. It is  
4 noted that for ease and clarity of illustration, no attempt has been made to illustrate in  
5 Figure 3 the well know fact that there is a difference in perspective between what is  
6 displayed in a view window and what is stored in a flat section of a rectilinear  
7 spherical panorama.

8  
9 Figure 2 is an overall diagram of a preferred embodiment of the invention. A camera  
10 unit 10 captures images. The images are sent to a computer 20 which stores the  
11 images. Computer 20 also controls camera unit 10. If desired the images can be  
12 viewed by a real time viewer 30. The images are transferred from computer 20 to off  
13 line computer 21. Computer 21 seams the images into panoramas, transforms the  
14 images to equirectangular format, adds other information to the images, compresses  
15 the panoramas, and links the panoramas into a pan movie. Finally the pan movie is  
16 viewed on viewer 22.

17  
18 The operations performed by the units in Figure 2 are shown in Figures 3A, 3B, 3C  
19 and 3D. As shown in Figure 3A, block 11a, camera unit 10 captures a number of  
20 single view images. As indicated by block 11b these images are compressed and  
21 sent to a computer 20. Computer 20 activates camera 10 to capture the images as  
22 indicated by block 20a. It then accepts the images as indicated by block 20b and  
23 stores them.

24  
25 The stored images are manually transferred to off line computer 21 which is  
26 programmed to perform the operations shown in Figure 3C. First the images are  
27 decompresses as indicated by block 20a so that they can be manipulated. Next the  
28 single view images are seamed into a panorama and transformed to equirectangular  
29 format as indicated by block 21b. Hot spots which indicate break points in a  
30 sequence of images and sound tracks are added next as indicated by block 21c.  
31 Finally the images are compressed as indicated by block 21d and stored with an  
32 index file as indicated by block 21e. Each panorama is termed a "key frame". A  
33 series of key frames (or more precisely a sequence of view windows) projected in  
34 sequence is a pan movie.

35



1 A viewer program in viewer computer 22 is used to view the pan movies. The viewer  
2 22 displays in sequence a series of images, that is, a series of key frames. For each  
3 key frame displayed the viewer 22 determines an appropriate view window as  
4 indicated by block 22a. The portion of the key frame which corresponds to the view  
5 window is then de-compressed and displayed as indicated by block 22b. As  
6 indicated by block 22c, sound is played if appropriate.

7  
8 It is noted that the operations indicated by blocks 20a, 20b, 21a to 21e, 22a, 22b, and  
9 22c are implemented by means of computer programs which perform the functions  
10 shown. Computer programs are given in appendices A, B, C, and D.

11  
12 Figure 4A represents or illustrates a sequence or series of panoramic images in a  
13 pan movie. Each arrow in Figure 4A represents one key frame. At any particular  
14 time, only a part (i.e. the view window) from one key frame is visible to a user or  
15 observer. The direction of each arrow indicates the direction of view, that is, the view  
16 window or part of the key frame that is projected on a screen for observation. The  
17 arrows in Figure 4A are meant to represent a particular "view window" from each key  
18 frame. As indicated by the change in direction of the arrows in the area of Figure 4A  
19 designated by the letter E, a viewer can change his direction of view as the pan  
20 movie progresses. It is noted that when a user is viewing a panorama, a user can  
21 point toward the top or bottom of the screen and thus can view images located in a  
22 360 degree circle from top to bottom in addition to the horizontal directions illustrated  
23 by the arrows shown in Figure 4A.

24  
25 The sequence of images begins at the point or at the key frame indicated by the  
26 letter A and the sequence proceeds to the point or key frame indicated by the letter  
27 B. At this point the viewer can select to either go toward point C or toward point D.  
28 The selection may be made by "clicking" on a designated "hot spot" in the panorama  
29 designated B or it may be made depending on some other criteria or action by the  
30 user. An important point is that at the branch point B, the direction of view (indicated  
31 by the direction of the arrows) remains the same irrespective of which path of travel  
32 is chosen. The view from the first frame after the branch point will be almost identical  
33 in both paths. As time progresses and the viewer moves further from the branch  
34 point, the view will gradually change. This is the effect that a person experiences

1 when one arrives at a dividing point in a path. When a person takes the first step on  
2 a branching path, the persons field of view remains practically identical.

3  
4 It is noted that at branch point B, the arrows are not pointing in the direction of the  
5 path leading to point D. Normally, a viewer would be looking in the direction of a  
6 branch point when the viewer selects to travel in the direction of the branch point.  
7 Thus, a viewer looking in the direction of the arrows shown in Figure 4A would  
8 normally continue to point C rather than selecting the path to point D.

9  
10 Sequences of key frames can either be joined at branch points such as branch point  
11 B or alternatively a branch point may be located at the end of a sequence of key  
12 frames. That is, a branch point may be located at the terminal frame of a sequence  
13 of key frames. Such a branch point could have two alternative sequences, one of  
14 which can be selected by a user by clicking on one of two hot spots. Alternatively at  
15 the end of a sequence of key frames, there can be an implicit branch point. At such  
16 an implicit branch point a new sequence of frames would be selected by the system  
17 without any action by the user.

18  
19 There is a one to one ratio of key frames to possible user positions. Hence, there  
20 exists a correlation between frame rate and user motion speed. If the user is moving  
21 through the environment, every frame displayed is a new key frame. The faster the  
22 frame rate for a given frame spacing, the faster the user travels. Given a fixed frame  
23 rate, the user's travel speed may be dictated by the relative spacing of key frames.  
24 The closer the key frames are, the slower the user will travel. For example, for a  
25 travel speed of approximately 5 mph and a playback frame rate of 15 fps, individual  
26 panoramic frames should be captured at about 6 inch increments. The math is as  
27 follows:  $(5 \text{ miles/hour} * 63,360 \text{ inches/mile}) / (3600 \text{ sec/hour} * 15 \text{ frames/sec}) = 6$   
28 inches per frame. When the movie is being displayed, speed of travel can be  
29 increased by skipping some of the frames (for example if every other frame is  
30 skipped the speed of travel is doubled). Skipping frames reduces the rate at which  
31 frames need be sent to the viewer and thus reduces the bandwidth required.

32  
33 In addition to the spacing of key frames to achieve different travel speeds, the  
34 orientation of individual key frames may be adjusted in order to achieve a desired  
35 motion effect, such as gate, slumber, waddle, crawl, skip, etc. The orientation of a

1 key frame is defined to be the default view (or point of focus) of the user within the  
2 panoramic image if no other point of view is specifically selected.

3  
4 Sound can accompany the visual effect provided by pan movies. Figure 4B indicates  
5 that each key frame can have one or more associated digital sound tracks. The  
6 digital sound tracks are indicated in Figure 4B by the dotted line which is associated  
7 with each of the arrows. As shown in Figure 11 and described later, there can be  
8 several different sound tracks associated with each key frame.

9  
10 Figure 5A is a perspective view of the six lens camera unit 10 which is used to  
11 digitally capture panoramic images, that is, key frames. Figure 5B is a top view of  
12 camera 10 which is included to show that the unit has six lenses 41a to 41f. Each  
13 lens 41a to 41f has a 110 degree field of view. The images captured by lenses 41a  
14 to 41f are transmitted to computer 20 through a serial connection. Computer 21  
15 "seams" the individual images from lenses 41a to 41f into panoramic images or key  
16 frames, compresses the key frames and stores them for future display. Additionally,  
17 a real time viewer can 30 can be used to view the images as they are being captured  
18 and seamed.

19  
20 In the preferred embodiment the connection from camera unit 10 to computer 20 and  
21 from computer 20 to real time viewer 30 is a "HOTlink" serial bus. Such connections  
22 are commercially available from suppliers such a Cypress Semiconductor Corp. or  
23 from Dataforth Corporation which is a division of Burr-Brow Company. Alternatively  
24 other types of high speed connections could be used. For example the connection  
25 could be a standard SCSI connection.

26  
27 Figure 6 shows the electronic components in camera 20. The components  
28 associated with each lens 41a to 41f are substantially identical to the components in  
29 commercially available digital cameras. The internal operation of the camera 20 is  
30 controlled by a conventional embedded programmed computer 45 which, for  
31 example, may be a model 29000 computer available from Advanced Micro Devices  
32 Corporation. Many different suitable embedded processors are commercially  
33 available. Embedded computer 45 receives commands from computer 20 which has  
34 IO units which allow an operator to enter commands. For example computer 20  
35 sends commands to computer 45 which set the aperture of the lenses 41a to 41f and

1 which starts and stops the operation of the camera. While computer 20 sends  
2 general commands to computer 45 such as set aperture, start, stop, etc., computer  
3 45 sends to detailed commands which control CCD arrays 43a to 43f and which  
4 control compression chips 44a to 44f. Such commands are conventional.

5  
6 Each of the lenses 41b to 41f has a set of associated components similar to the  
7 components associated with lens 41a.. The following will discuss the components  
8 associated with lens 41a. It should be understood that the other lenses 41b to 41f  
9 have a similar sets of components.

10  
11 The image from lens 41a is focused on a CCD (Charge Coupled Device) array 43a.  
12 CCD array 43a captures the image from lens 41a and send this image to embedded  
13 computer 45. CCD arrays 43a is controlled and operated by embedded computer  
14 45. By resetting and reading the CCD array 43a in a particular time period, the  
15 embedded computer 45 in effect controls or provides an electronic shutter for lens  
16 41a. The electronic shutters associated with each of the lenses 40a to 40f open and  
17 close simultaneously. Each CCD array 43a to 43f captures 30 images per second  
18 under normal operation.

19  
20 The output of CCD array 43a is fed into a JPEG data compression chip 44a. Chip  
21 44a compresses the image from lens 41a so that the image can be more easily  
22 transmitted to computer 40. The output from compression chip 41a fed to an  
23 embedded controller 45 which transmits signals to computer 40 on a serial time slice  
24 basis.

25  
26 The lenses 41 and the CCD arrays 43, and are similar to the components found in  
27 commercially available digital cameras. JPEG compression chips 44 and embedded  
28 computer 45 are also commercially available components. For example such  
29 components are available from suppliers such as Zoran Corporation or Atmel  
30 Corporation.

31  
32 The electronic shutters associated with lenses 41 operate at 30 cycles per second  
33 and hence computer 21 receives six images (one from each lens) each 1/30th of a  
34 second. The six images received each 1/30th of a second must be seamed and

1 transformed to equirectangular format to form one panorama as indicated by step  
2 21b in Figure 2.

3  
4 While the specific embodiment of the invention shown herein utilizes a digital camera  
5 to take the initial single view images which are compressed and sent to computer 20  
6 for storage, it should be understood that one could use a variety of other types of  
7 cameras to take these initial images. For example the images simultaneously taken  
8 from a number of lenses could be recorded on tape for later processing by off line  
9 computer 21.

10

11 The seaming operation is done by the program in computer 21. In general the  
12 seaming operation connects the individual images into a panoramic image by finding  
13 the best possible fit between the various individual images. The process of seaming  
14 images into a panoramic image is known. For example U.S. patent 5,694,531  
15 describes seaming polygons into a panorama which has a low root-mean-square  
16 error.

17

18 After the seaming operation is complete each seamed image is a panoramic image  
19 (called a panorama) and each panorama is a frame of a pan movie. Prior to storage  
20 the seamed images are compressed so as that the file size will be manageable. A  
21 commercially available compression program known as "Indeo" is used to compress  
22 the images. The Indeo program was developed by and is marketed by the Intel  
23 Corporation. The Indeo compression program provides a mode of operation which  
24 does not utilize any inter-frame compression. The no inter-frame compression mode  
25 of the Indeo program is used with the present embodiment of the invention. Since  
26 there is no inter frame compression, the key frames can be accessed and viewed in  
27 either the forward or the reverse direction. Furthermore, only the portion of a  
28 panorama required for a particular view window is decompressed, thereby saving  
29 time and computational resources.

30

31 The compressed panoramic images are stored in files on computer disks, tape or  
32 compact discs (CDs). Each file includes a header and an index as shown in Figure 7.  
33 The header includes information such as the following:

34 File Type Tag:

35 File Size: (total bytes used by the file)

1           Index Size:           (Number of entries in frame Index)  
2           Max Frame Size:       (total bytes used by largest compressed frame)  
3           Codec:                (Codec used to compress frames.  
4   After the file header, a frame index is provided (see Figure 7). Each frame index  
5   points to the location of the associated frame as indicated by the arrows in Figure 7.  
6   Thus, individual frames can be read in any order by obtaining their location from the  
7   frame index.  
8  
9   The indexing mechanism would not be necessary if the key frames were always  
10   going to be used in frame order. However, in the present embodiment, the system  
11   can play the key frames which comprise the pan movie in either forward or backward  
12   direction. Hence the system must be able to locate individual frames quickly in any  
13   order. Furthermore, it is desirable that the system be able to locate a key frame with  
14   only a single disk access. Consider the situation were the user is moving "backward"  
15   (in the opposite direction of the key frame disk storage) at a fast travel speed (to  
16   increase speed of movement some key-frames are skipped). Without a key frame  
17   directory, the disk would have to be searched in a "reverse-linear" manner in order to  
18   find and load the next appropriate key frame. With a key frame directory, the next  
19   key frame location is located immediately, and loaded with a single disk access  
20   (given the directory itself is stored in RAM memory).  
21  
22   As indicated in Figure 4A, a viewer can branch from one sequence of images to  
23   another sequence of images. This is indicated by branch point B in Figure 4A. By  
24   branching a user in effect changes the direction of the simulated travel. A user  
25   indicates a desire to change direction by "clicking" on a visible "hot spot" or by  
26   otherwise activating a hidden hot spot. A visible hot spot can be indicated by any  
27   type of visible symbol that is visible in a view window. For example a hot spot may  
28   be indicated by a bright red dot in the view window. Alternatively, a hot spot may be  
29   indicated by the fact that the cursor changes to a different shape when the cursor is  
30   over a hot spot.  
31  
32   It is noted that not all visually apparent alternate paths visible in any panorama are  
33   actually available as a pan movie branch. For example, at a street intersection,  
34   branches may not be provided to all visible streets. Care must be taken to insure

1 that a viewer is given an indication of the branch points that are actually available to  
2 the viewer.

3  
4 At a playback rate of 30 frames per second a user would have to be very "fast" (i.e. it  
5 would in fact be practically impossible) for a viewer to see and click on a hot spot that  
6 appears on a single frame. Without advanced notice, the viewer would have great  
7 difficulty actually taking a specific action to activate a branch during a specific single  
8 frame since in normal operation a particular frame is only displayed for about 1/30th  
9 of a second. In order to be effective and user friendly a user must be given an early  
10 indication of a upcoming branch opportunity that requires user action. A hot spot in a  
11 pan movie must be visible by a viewer in a relatively large number of key frames. For  
12 example a hot spot might be visible in the thirty key frames that precede (or follow for  
13 reverse operation) a branch point.

14  
15 Hot spots are inserted into a pan movie in the manner illustrated in Figure 8. The hot  
16 spots are inserted into the key frames by computer 21 before the frames are  
17 compressed as indicated by blocks 21c and 21d in Figure 3C. It is noted that hot  
18 spots may be inserted into a pan movie by altering the original panoramic image so  
19 that it includes the hot spot or alternately by providing a overlay image which  
20 contains the hot spot image. If an overlay is used, the overlay image needs be  
21 projected at the same time as the original image. As indicated by block 87a one  
22 must first determine how much in advance one wants to warn the user. If a hot spot  
23 is to have a particular size at the time action is needed, when viewed in advance (i.e.  
24 from a distance) the hot spot will be much smaller. As indicated by block 87b, in  
25 order to insert hot spots in a pan movie, one must select the region where the hot  
26 spot is to be located. In general this will be in a view looking toward the direction  
27 where the branch will take place. The hot spot is then inserted into the panorama by  
28 modifying the images. The hot spot may be indicated by a light colored outline  
29 superimposed over the region. The area within the outline may be slightly darkened  
30 or lightened. The object is to highlight the region without obscuring the image itself.  
31 Various other alternative indications can also be used.

32  
33 The process repeats as indicated by blocks 87d and 87e until the key frame at the  
34 branch point is reached. Finally the process is repeated from the opposite direction

1 from the branch point so that the branch point will be visible if the pan movie is  
2 shown in the reverse direction.

3  
4 The changes to the individual key frames may be made manually with a conventional  
5 image editor, or the process can be automated by a program designed just for this  
6 purpose.

7  
8 In order to avoid unnecessary user intervention, "hidden" hot spots may be added to  
9 connect multiple pan movies. A hidden hotspot is one that does not need to be  
10 manually selected by the user. With a hidden hot spot, if the user "travels" into a  
11 particular key frame which has a hidden hot spot, and the user is "looking" in the hot  
12 spot's general direction, then the system will react based upon the user's implicit  
13 selection of the hotspot and the user will be sent along the path directed by the hot  
14 spot.

15  
16 Figure 9A is a block diagram of the viewer 22 which plays or displays pan movies.  
17 The main components of the viewer 22 are a CD disk reader 80, a computer 81, a  
18 display 82, a keyboard 84 and a mouse 85. Computer 81 reads key frames from disk  
19 80 and displays the view widow from each key frame on display 82. The operator or  
20 user utilizes mouse 85 to indicate a view direction. The view direction determines the  
21 view window which is displayed on display 82 by computer 81. A program which  
22 implements blocks 22a to 22c (shown in Figure 3D) is stored in and executed by  
23 computer 81.

24  
25 Figure 9B is a block diagram of the real time viewer 30. As an option, the images  
26 captured by camera 10 can be viewed in real time. Images are transferred from  
27 computer 21 to viewer 22 in real time. The transfer is by means of a HOTlink bus to  
28 HOTlink card 86a. The images go from card 86a to RAM memory 86b and then to  
29 decompression card 86c which does the de-compression. From the de-compression  
30 board 86c the images go back to memory and then to CPU 86d which combines i.e.  
31 seams the images as necessary and transfers them to video card 86e which displays  
32 them on monitor 86f. Viewer 30 is controlled via a conventional mouse 86m and  
33 keyboard 86k.

34



1 Figure 10 is block diagram of a program for displaying pan movies. The program  
2 shown in block diagram in Figure 10 is executed by the computer 81 in Figure 9A.  
3 The process begins at block 91 with user input. The user must indicate a start  
4 location (at the beginning of the process this would normally be the first frame in the  
5 movie). The user must also specify direction of motion, speed and direction of view.  
6 As indicated by blocks 92, 92a, 92b and 92c the system determines and then reads  
7 the appropriate pan frame data. As indicated by block 96 and 96a, the system  
8 determines the portion of the pan frame that is in the selected view window and that  
9 portion of the frame is decompressed. As indicated by blocks 97 and 97a, the image  
10 is re-projected to obtain a perspective view. If the hot spots have not been placed on  
11 the actual key frames but are contained in a separate file, the hot spot imagery is  
12 overlaid on the image. Finally, as indicated by block 98, the part of the image which  
13 constitutes the view window is projected on the screen.

14  
15 As a user travels, the next required key frame is determined by the current user  
16 position and direction of travel. The location of this key frame within the file of  
17 images is determined via the file index directory; and the key frame is loaded into  
18 RAM memory, decompressed, and displayed. To increase performance, only the  
19 view window (depending on current user view) portions of the key frame need be  
20 loaded into RAM. If for ease of programming the entire key frame is loaded into  
21 memory, only view window portions of the key frame need be decompressed. If the  
22 entire key frame is compressed as a whole, then a de-compressor supporting "local  
23 decompression" is more efficient, e.g., Intel Indeo. To determine the portion of the  
24 panorama needed to display a particular view, each of the corner coordinates of the  
25 perspective view plane (display window) is converted to panorama coordinates. The  
26 resulting panorama coordinates do not necessarily represent a rectangle, therefore  
27 the bounding rectangle of these panorama data is needed to derive a perspective  
28 view at a given view orientation.

29  
30 Once the corners of the desired bounding rectangle are determined the Indeo de  
31 compression program is instructed to decompress only that portion of the key frame  
32 needed for the particular view window. In order to do this, the program must call the  
33 Video For Windows function ICSetState prior to decompressing the frame. The C  
34 code to accomplish this follows.

35

```

1  #include "windows.h"
2  #include "vfw.h"
3  #include "vfw_spec.h"
4
5  extern HIC          hic;          // Opened CODEC (IV41);
6  extern RECT         *viewRect;    // Determined elsewhere
7  static R4_DEC_FRAME_DATA  StateInfo;
8
9  void  SetRectState
10 (
11     HIC  hic;          // Opened CODEC (IV41);
12     RECT *viewRect;    // Local Rectangle of interest
13 )
14 {
15     R4_DEC_FRAME_DATA  StateInfo;
16
17     memset(&StateInfo,0,sizeof(R4_DEC_FRAME_DATA));
18     StateInfo.dwSize = sizeof(R4_DEC_FRAME_DATA);
19     StateInfo.dwFourCC = mmioStringToFOURCC("IV41",0); // Intel Video 4.1
20     StateInfo.dwVersion = SPECIFIC_INTERFACE_VERSION;
21     StateInfo.mtType = MT_DECODE_FRAME_VALUE;
22     StateInfo.oeEnvironment = OE_32;
23     StateInfo.dwFlags = DECFRAME_VALID | DECFRAME_DECODE_RECT;
24
25     StateInfo.rDecodeRect.dwX = min(viewRect->left,viewRect->right);
26     StateInfo.rDecodeRect.dwY = min(viewRect->top,viewRect->bottom);
27     StateInfo.rDecodeRect.dwWidth = abs((viewRect->right-viewRect->left)+1);
28     StateInfo.rDecodeRect.dwHeight = abs((viewRect->bottom-viewRect-
29 >top)+1);
30
31     ICSetState(hic,&StateInfo,sizeof(R4_DEC_FRAME_DATA));
32 }
33
34 If the projection used to store the pan-frame is such that there exists a discontinuity
35 in pixels with respect to the spherical coordinates they represent, then the local
36 region required may be the combination of multiple continuous regions. For a full

```

cylinder/sphere equirectangular projection (centered about 0 degrees), the left pixel edge represents -180 degrees and the right pixel edge represents 180 degrees. In spherical coordinates, -180 degrees is the same as 180 degrees. Therefore, the discontinuous left/right pixels represent a continuous "wrap-around" in spherical coordinates.

The math to determine the portion of the source key-frame panorama needed for a particular view window depends on the projection used to store the panorama. Optionally, the viewer may predict the next key-frame to be loaded (depending on user travel direction and speed), and pre-load it in order to increase performance. For an equirectangular projection of a full sphere panorama frame, the equations for determining the required portion are as follows:

where:

Scalar variables are lower case, vectors are bold lower case, and matrices are bold upper case.

Panorama point (s,t) is derived from any perspective plane point (u,v).

The perspective plane has a focal length  $l$  from the center of projection.

In addition, the perspective plane can be arbitrarily rotated through a given view orientation, namely heading, pitch, and bank (h,p,b).

Any point in the perspective plane is specified by the 3D vector:

$$\mathbf{w} = \langle u, v, l \rangle$$

The rotations are applied by using a standard matrix-vector product. The three matrices accounting for Heading, Pitch and Bank are as follows:

$$\mathbf{H} = \begin{bmatrix} \cos(h) & 0 & \sin(h) \\ 0 & 1 & 0 \\ -\sin(h) & 0 & \cos(h) \end{bmatrix}$$

$$\mathbf{P} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(p) & -\sin(p) \\ 0 & \sin(p) & \cos(p) \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \cos(b) & \sin(b) & 0 \\ -\sin(b) & \cos(b) & 0 \end{bmatrix}$$

1                   | 0    0    1                   |

2

3           The vector **w** is rotated using the above matrices to attain **w'** like such"

4           **w' = H\*P\*B\*w**

5   The final step is converting from rectangular to spherical coordinates. Denoting the 3  
6   components of the vector **w'** as x, y, z, then the conversion is:

7           s = atan2(x, z)

8           t = atan2(y, sqrt(x\*x + z\*z))

9   Note: atan2(a, b) is a standard C-function very similar to atan(a/b), but atan2  
10   correctly handles the different cases that arise if a or b is negative or if b is 0.

11

12   Optionally, the viewer may predict the next key-frame to be loaded (depending on  
13   user travel direction and speed), and pre-load this key frame in order to increase  
14   performance.

15

16   Due to the one to one ratio of key frames to possible user positions, there exists an  
17   exact correlation between frame rate and user motion speed. If the user is currently  
18   moving through the environment, every frame displayed is a new key frame, thus the  
19   faster the frame rate, the faster the user travels. For this reason, the frame rate is  
20   "capped" during user travel to eliminate the problem of excessive user travel speed.  
21   In order to retain smooth motion, the frame rate is not decreased to below standard  
22   video frame rates (15 frames/sec.) The frame rate is not increased in order to keep  
23   the relative spacing of key frames to a manageable distance; the faster the frame  
24   rate, the closer the key frames must be to achieve the same user travel speed. The  
25   viewer may optionally skip key-frames in order to increase the user's travel speed  
26   through the environment. The more key-frames skipped, the faster the user will  
27   travel; if no key-frames are skipped, the user will travel at the slowest possible rate  
28   (given a constant frame rate.)

29

30   The system can link pan movie segments so as to permit branching and thereby  
31   follow a path selected by a user. Multiple linear (one dimensional) pan movies may  
32   be linked together to create a "graph" of pan movies (see appendix B). For each pan  
33   movie, the end of one segment may be associated with the start of a "next" pan  
34   movie. This association (in conjunction with the length of the individual pan movies)  
35   is the basis for the graph shape. In order to achieve smooth transitions, the "last"

1 frame in the "first" pan movie must be the same as (or one frame off from) the "first"  
2 frame of the "next" pan movie. In addition to positional correctness, the relative view  
3 orientations of the joining frames must be known. For example, if the "last" frame of  
4 the "first" pan movie faces "north", and the "first" frame of the "next" Pan Movie faces  
5 "east", then the viewing software must be alerted to this orientation change. Without  
6 this information, there would be a 90 degree "snap" in the transition between the two  
7 Pan Movies. All this graph information may be stored in a separate file (text or binary  
8 form.)

9  
10 The audio information associated with each frame of a pan movie must take into  
11 account the fact that a viewer of a pan movie has a great deal of control over what is  
12 presented on the screen. In addition to the ability to select branch points a user may  
13 choose to change the direction of view or to stop and backup. The audio information  
14 associated with each key frame must accommodate this flexibility.

15  
16 As illustrated in Figure 11, the audio information stored with each key frame includes  
17 five audio tracks designated A, B, C, D, E and control information. Figure 11 shows  
18 eight key frames Fa to Fi each of which has five associated audio tracks and a  
19 control field. Audio track A is the track that is played if the pan movie is moving  
20 forward in the normal direction at the normal rate of thirty frames per second. Audio  
21 track B is the track that is played if the pan movie is being displayed in reverse  
22 direction. Audio track C is the audio track that is played if the movie is moving  
23 forward at half speed. Audio track D is the track that is played if the movie is being  
24 played in the reverse direction at one half speed. Finally audio track E is the track  
25 that is repeatedly played if the movie has stopped at one frame. Naturally a variety  
26 of other audio tracks could be added for use in a number of other situations. For  
27 example, tracks can point to audio clips or to other audio tracks.

28  
29 The control information that is recorded with each frame controls certain special  
30 effects. For example the control information on one frame can tell the program to  
31 continue playing the audio tracks from the following frame even if the user has  
32 stopped the movie at one particular frame. As the sound track on each frame is  
33 played, the control information on that frame is interrogated to determine what to do  
34 next. What sound is played at any particular time is determined by a combination of  
35 the control information on the particular frame being viewed and the action being

1 taken by the viewer at that time. From a programming point of view, the commands  
2 associated with each rack are de-compressed and read when the view window for  
3 the associated frame is de-compressed and read. As a particular view window is  
4 being displayed (or slightly before) the commands stored in the control field are read  
5 and executed so that the appropriate sound can be de-compressed and played when  
6 the view window is displayed.

7  
8 For example the control information could provide the following types of commands:

9 Stop this audio track if user stops pan movie here (typical setting). If this is  
10 not set the audio will continue playing in same direction until audio for this  
11 track ends

12  
13 Start or continue to play this audio track if user is viewing pan movie in  
14 forward direction (typical setting)

15  
16 Start or continue to play this audio track backwards if user if viewing pan  
17 move in a backwards direction. (note if the same audio information is played  
18 is reverse it may be distorted)

19  
20 Start this audio track when image frames are in motion and being played in a  
21 reverse direction. This allows high quality audio to be played while reverse  
22 viewing

23  
24 Continue audio track from / on other file structure (branch most likely has  
25 occurred) modify volume This is used to fade out an audio track that may  
26 have played ahead earlier

27  
28 Stop all audio tracks

29  
30 Stop this audio track if user slows pan movie playback

31  
32 Start audio file X: where X is a conventional audio file that is separate from  
33 the pan movie.

1           A wide variety of other commands may be implements as desired by the  
2           designer of a particular movie.

3  
4           The audio information can be recorded with a normal recorder when the initial  
5           images are recorded or it can be recorded separately. The audio data is merged with  
6           the key frames by computer 21. This can be done manually on a frame by frame  
7           basis or the process can be automated. When the sound is merged with the key  
8           frames the appropriate control information is added.

9  
10          Figures 12A, 12B and 12C illustrate another aspect of the present invention. A hot  
11          spot on a key frame can contain and display information that is independent from,  
12          and in addition to the information in the base images which are used to form the  
13          panoramas. For example, as a pan movie simulates movement past a billboard, a  
14          "regular" motion picture (which might for example be an advertisement for a product)  
15          can be displayed on the billboard. The motion picture on the billboard would be  
16          integrated with the various key frames in the same manner as hot spots are added to  
17          key frames. As illustrated in Figure 12A, 12B and 12C, such images displayed on a  
18          billboard passed which motion is simulated must be corrected for the fact that the  
19          viewer is not directly viewing the image when he is approaching it. The image is only  
20          rectangular when the viewer is adjacent the image as shown in Figure 12C. As the  
21          viewer is approaching the image it is distorted as illustrated in Figures 12A and 12B.

22  
23          The attached appendices provide computer programs which implement various  
24          aspects of the present invention. These programs are designed to run under a  
25          conventional operating system such as the "Windows" operating system marketed by  
26          the Microsoft Corporation.

27  
28          The program given in Appendix A will retrieve frames for a move, correct the  
29          perspective in accordance with known equations and then display the images of the  
30          movie in sequence.

31  
32          Appendix B is an example of a link control file for the frames of a pan movie.  
33          Appendix C is pseudocode showing how sequences of images are linked to form a  
34          pan movie.

35

1 It is noted that in a pan movie the frames do not all have to have the same resolution.  
2 Some frames may be of a higher resolution. For example, at the most interesting  
3 places in the Pan Movie may have a higher resolution.

4  
5 Many alternative embodiments of the invention are possible. For example, the initial  
6 capture process could record the images on video tape rather than recording the  
7 images digitally. Electronic cameras could be used which include image capture  
8 devices other than CCD arrays to capture images. Branching can provide three or  
9 more optional paths rather than just two paths as shown in Figure 4; and branching  
10 can provide for going left or right at an intersection.

11

12 It is noted that in alternative embodiments, compression schemes or techniques  
13 other than Intel Indio can be used. Furthermore, alternative embodiments could use  
14 no compression at all if enough storage and bandwidth were available.

15

16 While in the embodiment shown the images files are manually transferred between  
17 some of the units in the embodiment shown, in alternative embodiments these files  
18 could be transferred between units by electronic connections between the units.

19

20 While in the embodiment described above the camera has six lenses which record all  
21 six sides of the cube, in alternative embodiments the camera could record less than  
22 an entire sphere. For example the lens pointing down could be eliminated in some  
23 embodiments. Still other alternative embodiments could use lenses with wider or  
24 narrower fields of view. For example less lenses each with a wider field of view could  
25 be used. Furthermore, while the embodiment described above utilizes spherical  
26 panorama, other types of panoramas could be used. Various types of projections  
27 such as cubic could be used instead of equi-rectangular.

28

29 The embodiment shown includes a number of sound tracks with each key frame and  
30 control information which indicates which sound track should be played when the key  
31 frame is displayed depending on whether or not certain special conditions exist.  
32 Alternatively, there could be a single sound track associated with each frame. In  
33 such an embodiment the single sound track on each key frame could be the sounds  
34 recorded when the images in the particular frame were recorded. In other



1 alternatively embodiments, there could be no sound tracks and in such case the  
2 images would be displayed without accompanying sound.

3

4 Having described and illustrated the principles of the invention in various  
5 embodiments thereof, it should be apparent that the invention can be modified in  
6 arrangement and detail without departing from the principles of the invention. We  
7 claim all modifications and variation coming within the spirit and scope of the  
8 following claims.

9

10

11

12

```

1      APPENDIX A: FRAME RETRIEVAL CODE
2
3      #include "windows.h"
4      #include "mmsystem.h"
5      #include "vfw.h"
6      #include "vfw_spec.h"
7
8      #define S_BMIH      sizeof(BITMAPINFOHEADER)
9
10     // Externally declared (and allocated) variables
11     extern UINT currentFrameNumber; // Current Pan Movie file frame number
12     (user position)
13     extern HANDLE hFile; // Open file handle of Pan Movie file
14     extern HIC hic; // Open IC handle (installed compressor)
15     extern DWORD *Index; // Pan Movie Frame Index (read from file at load
16     time)
17     extern LPBITMAPINFOHEADER viewFrame; // Buffer large enough to hold
18     image the size of the display window
19     extern LPBITMAPINFOHEADER panFrame; // Buffer large enough to hold
20     largest uncompressed frame
21     extern LPBITMAPINFOHEADER compressedFrame; // Buffer large enough to
22     hold largest compressed frame
23
24     // Function prototypes
25     extern void ViewToPan(int viewWidth,int viewHeight,int panWidth,int
26     panHeight,float heading,float pitch,float bank,float zoom,POINT *point);
27     static LPBITMAPINFOHEADER RetrievePanFrame(int frameNumber,RECT
28     *viewRect);
29
30     //
31     // This function generates a perspectively correct bitmap image given a
32     user view orientation and travel speed
33     //
34     static LPBITMAPINFOHEADER RetrieveViewFrame(float userHeading,float
35     userPitch,float userBank,float userZoom,int userTravelSpeed)
36     {
37         // Determine Decode BoundingBox
38         POINT point;
39         RECT localDecompressionRect;
40
41         // Upper left corner of viewFrame
42         point.x = 0; point.y = 0;
43
44         ViewToPan(viewFrame->biWidth,viewFrame->biHeight,panFrame->biWidth,panFrame->biHeight,
45         userHeading,userPitch,userBank,userZoom,&point);
46         localDecompressionRect.top = point.y;
47         localDecompressionRect.left = point.x;
48
49         // Upper right corner of viewFrame
50         point.x = viewFrame->biWidth-1; point.y = 0;
51
52         ViewToPan(viewFrame->biWidth,viewFrame->biHeight,panFrame->biWidth,panFrame->biHeight,
53         userHeading,userPitch,userBank,userZoom,&point);
54         localDecompressionRect.top = min(localDecompressionRect.top,point.y);
55         localDecompressionRect.right = point.x;
56

```

```

1      // Lower left corner of viewFrame
2      point.x = 0; point.y = viewFrame->biHeight-1;
3
4      ViewToPan(viewFrame->biWidth,viewFrame->biHeight,panFrame->biWidth,panFrame->biHeight,
5      userHeading,userPitch,userBank,userZoom,&point);
6      localDecompressionRect.bottom= point.y;
7      localDecompressionRect.left    =
8      min(localDecompressionRect.left,point.x);
9
10     // Lower right corner of viewFrame
11     point.x = viewFrame->biWidth-1; point.y = viewFrame->biHeight-1;
12
13     ViewToPan(viewFrame->biWidth,viewFrame->biHeight,panFrame->biWidth,panFrame->biHeight,
14     userHeading,userPitch,userBank,userZoom,&point);
15     localDecompressionRect.bottom=
16     max(localDecompressionRect.bottom,point.y);
17     localDecompressionRect.right  =
18     max(localDecompressionRect.right,point.x);
19
20     // Get Pan Frame (or "userDecompressionRect" portion thereof)
21     currentFrameNumber += userTravelSpeed; // userTravelSpeed is negative
22     if traveling backwards
23         LPBITMAPINFOHEADER pFrame =
24         RetrievePanFrame(currentFrameNumber,&localDecompressionRect);
25
26         if(pFrame == NULL) {
27             currentFrameNumber -= userTravelSpeed;
28             return NULL;
29         }
30
31         // A very slow warping routine (assumes 24-bit pixels)
32         LPBYTE srcPixels = ((LPBYTE)pFrame) + S_BMIH;
33         LPBYTE dstPixels = ((LPBYTE)viewFrame) + S_BMIH;
34         for(int y = 0; y < viewFrame->biHeight; y++) {
35             for(int x = 0; x < viewFrame->biWidth; x++) {
36                 point.y = y; point.x = x;
37
38                 ViewToPan(viewFrame->biWidth,viewFrame->biHeight,pFrame->biWidth,pFrame->biHeight,
39                 userHeading,userPitch,userBank,userZoom,&point);
40                 memcpy(&dstPixels[3*(x +
41 y*viewFrame->biWidth)],&srcPixels[3*(point.x
42 + point.y*pFrame->biWidth)],3); // supports 24-Bit Pixels only
43             }
44         }
45
46         return viewFrame;
47     }
48
49     //
50     // This function reads and decompresses a Pan Frame bitmap image from a
51     Pan Movie file
52     //
53     static LPBITMAPINFOHEADER RetrievePanFrame(int frameNumber,RECT
54     *viewRect)
55     {
56         DWORD d;
57         UINT frameSize= Index[frameNumber+1]-Index[frameNumber];

```

```

1
2      // Set the file pointer to the start of the requested frame and read in
3  the bitmap header
4      SetFilePointer(hFile, Index[frameNumber], NULL, FILE_BEGIN);
5      ReadFile(hFile, panFrame, S_BMIH, &d, NULL);
6
7      if(panFrame->biCompression == 0) {    // Uncompressed frame (read rest of
8  frame and return)
9          ReadFile(hFile, ((BYTE*)panFrame)+S_BMIH, frameSize-S_BMIH, &d, NULL);
10         return panFrame;
11     }
12
13     // Read the remainder of the compressed frame
14     *compressedFrame = *panFrame;
15
16     ReadFile(hFile, ((BYTE*)compressedFrame)+S_BMIH, frameSize-S_BMIH, &d, NULL);
17
18     // Set up decompressed bitmap header
19     panFrame->biCompression = 0;
20     panFrame->biSizeImage = 0;
21     panFrame->biBitCount = 24;
22     panFrame->biClrUsed = 0;
23
24     LPBITMAPINFOHEADER biSrc = compressedFrame;
25     LPBITMAPINFOHEADER biDst = panFrame;
26     LPBYTE srcPixels = (BYTE*)biSrc + S_BMIH;
27     LPBYTE dstPixels = (BYTE*)biDst + S_BMIH;
28
29     // If the frame is compressed with Intel Indeo 4 and a local rect was
30     requested, then perform local decompression
31     if(viewRect && biSrc->biCompression == mmioFOURCC('i','v','4','1')) {
32     // Intel Indeo 4.1
33         R4_DEC_FRAME_DATA    StateInfo;
34
35         memset(&StateInfo, 0, sizeof(R4_DEC_FRAME_DATA));
36         StateInfo.dwSize = sizeof(R4_DEC_FRAME_DATA);
37         StateInfo.dwFourCC = biSrc->biCompression;
38         StateInfo.dwVersion = SPECIFIC_INTERFACE_VERSION;
39         StateInfo.mtType = MT_DECODE_FRAME_VALUE;
40         StateInfo.oeEnvironment = OE_32;
41         StateInfo.dwFlags = DECFRAME_VALID |
42     DECFRAME_DECODE_RECT;
43         StateInfo.rDecodeRect.dwX = min(viewRect->left, viewRect->right);
44         StateInfo.rDecodeRect.dwY = min(viewRect->top, viewRect->bottom);
45         StateInfo.rDecodeRect.dwWidth =
46     abs((viewRect->right-viewRect->left))+1;
47         StateInfo.rDecodeRect.dwHeight =
48     abs((viewRect->bottom-viewRect->top))+1;
49
50         ICSetState(hic, &StateInfo, sizeof(R4_DEC_FRAME_DATA));
51
52
53     if(ICDecompressEx(hic, 0, biSrc, srcPixels, 0, 0, biSrc->biWidth, biSrc->biHeight, biDst, dstPixels, 0,
54     0, biDst->biWidth, biDst->biHeight)
55     != ICERR_OK )
56         return NULL;
57     }

```

```
1         else { // Decompress entire frame
2
3         if(ICDecompressEx(hic,0,biSrc,srcPixels,0,0,biSrc->biWidth,biSrc->biHeight,biDst,dstPixels,0,
4         0,biDst->biWidth,biDst->biHeight)
5         != ICERR_OK )
6             return NULL;
7         }
8
9         return panFrame;
10    }
11
12
13
14
15
16
17
18    © Infinite Pictures 1998
```

## APPENDIX B: SAMPLE PAN MOVIE LINK CONTROL FILE

```

1
2
3
4  <----->
5      <- C   |   B ->
6              |
7              |
8              |
9              | A
10             |
11             |
12
13 [Segment-A (start)]
14 File=      "A.pan"
15 North=      0
16
17 [Segment-A (end)]
18 File=      "A.pan"
19 North=      0
20 Link 90=    "Segment-B (start)"
21 Link 270=   "Segment-C (start)"
22
23 [Segment-B (start)]
24 File=      "B.pan"
25 North=     90
26 Link 90=    "Segment-A (end)"
27 Link 180=   "Segment-C (start)"
28
29 [Segment-B (end)]
30 File=      "B.pan"
31 North=     90
32
33 [Segment-C (start)]
34 File=      "C.pan"
35 North=    270
36 Link 270=   "Segment-A (end)"
37 Link 180=   "Segment-B (start)"
38
39 [Segment-C (end)]
40 File=      "C.pan"
41 North=    270
42
43 © Infinite Pictures 1998

```

```

1  APPENDIX C PSEUDOCODE FOR
2    LINKED PAN MOVIES (VIA CONTROL FILE)
3
4  GLOBAL FILE controlFile // Control file
5  GLOBAL STRING currentSegment // The name of the current pan movie
6  segment
7  GLOBAL INTEGER currentFrameNumber // The current frame number of the
8  current Pan Movie
9  GLOBAL      INTEGER currentHeading // The current user view horizontal pan
10 orientation
11
12 //
13 // This function will read the control file and determine which linked
14 segment is closest
15 // to the current user heading orientation
16 // It will also determine the new frame number of the new segment
17 //
18 BOOLEAN      RetrieveLink()
19 {
20     INTEGER minAngle
21     STRING nextSegment
22
23     if currentFrameNumber == 0
24         currentSegment = currentSegment + (start)
25     else
26         currentSegment = currentSegment + (end)
27
28     if no links in section currentSegment of controlFile
29         return FALSE
30
31     minAngle      = link angle closest to currentHeading
32     nextSegment   = GetString(minAngle)
33
34     if AngleDifference(currentHeading,MinAngle) > 45 degrees
35         return FALSE;
36
37     INTEGER nextNorth = GetNorth(nextSegment)
38     INTEGER currentNorth = GetNorth(currentSegment)
39
40     currentHeading = currentHeading + (nextNorth - currentNorth)
41     currentSegment = nextSegment
42
43     if stringFind(currentSegment,"(end)")
44         currentFrameNumber = -1
45     else
46         currentFrameNumber = 0
47
48     return TRUE
49 }
50
51
52  © Infinite Pictures 1998

```

1 I claim:

2

3 1) A system for simulating movement through multidimensional space comprising in  
4 combination,  
5 a multi-lens camera for simultaneously capturing a plurality of digital images that  
6 cover the entire spherical view field,  
7 compression units for individually compressing said images into compressed images,  
8 means for transferring said images to a computer,  
9 a program operated by said computer which seams said images into panoramas and  
10 which links said images into a sequence of images,  
11 a viewer which displays said linked images in sequence.

12

13 2) The system recited in claim 1 wherein said camera has six lenses, one positioned  
14 on each side of a cube.

15

16 3) The system recited in claim 1 wherein said computer compresses said images after  
17 seaming the images.

18

19 4) The system recited in claim 3 wherein said compression is single frame  
20 compression with no inter-frame compression.

21

22 5) A method of simulating movement through multidimensional space comprising the  
23 steps of:  
24 capturing a series of sets of individual images, each set of images covering at least a  
25 portion of a spherical view,  
26 individually compressing said images,  
27 transferring said images to a computer,  
28 decompressing said images,  
29 seaming the images in each set of images into a panorama,  
30 compressing said panoramas without using any inter-frame compression,  
31 de-compressing at least a portion of each panorama which corresponds to a view  
32 window, and  
33 displaying said view windows in sequence.

34



- 1 6) The method recited in claim 5 wherein each set of images comprises six images  
2 taken from the six sides of a cube.  
3
- 4 7) The method recited in claim 5 where the portion of each panorama de-  
5 compressed for viewing is selected by an operator who indicates a direction of view.  
6
- 7 8) The method recited in claim 5 wherein said sequence of images has break points  
8 which provide at least two alternative sequences of images.  
9
- 10 9) The method recited in claim 5 wherein said the sequence of images displayed  
11 after a break point is selected in response to input from a user.  
12
- 13 10) The method recited in claim 5 wherein sound is associated with each of said  
14 panoramas.  
15
- 16
- 17 11) A three dimensional (3-D) panorama movie for enabling a user interactively to  
18 view movement through a three-dimensional space along a path through a series of  
19 viewpoints and to view in any viewing direction in the three-dimensional space, the 3-  
20 D panorama movie comprising:  
21 a computer storage medium for storage of machine readable image data;  
22 a file of machine readable image data stored on the storage medium, the  
23 image data including a plurality of panorama frames forming a sequence of images of  
24 the three dimensional space in which each image in the sequence has a spatially  
25 different viewpoint; and  
26 a panframe directory stored on the storage medium in association with the  
27 file of machine readable image data, containing a set of frame indexes, each frame  
28 index identifying a location in the file of one of the panorama frames.  
29
- 30 12) A panorama movie according to claim 11 in which there exists a direction of  
31 travel in three-dimensional space from each viewpoint, the direction of travel being  
32 stored in the panorama frame associated with each viewpoint as a predetermined  
33 point within the panorama frame to define a frame of reference for a viewing direction  
34 during playback.  
35

1 13) A panorama movie according to claim 12 in which the predetermined point is  
2 a centerpoint of the panorama frame.

3  
4 14) A panorama movie according to claim 11 in which the panorama frames are  
5 compressed using intraframe compression, defining a key frame, so that each frame  
6 can be decompressed during playback independently of each other frame.

7  
8 15) A panorama movie according to claim 11 in which the file includes a hot spot  
9 associated with at least one of the frames, the hotspot being operable during  
10 playback to superimpose a function on the playback of that frame.

11  
12 16) A panorama movie according to claim 15 in which the hotspot includes an image  
13 having a predetermined geometric shape to be superimposed over a feature in one  
14 or more of the panorama frames.

15  
16 17) A panorama movie according to claim 15 in which the hotspot image is stored  
17 with an orientation in each panorama frame such that the superimposed image  
18 appears in each frame at playback with a position and shape that conforms to a  
19 perspective corrected shape and position of the feature over which it is  
20 superimposed.

21  
22 18) A panorama movie according to claim 11 which includes two or more of said  
23 files, each forming a segment of the movie and having its own respective panframe  
24 directory which includes terminal file indices defining a firstfile index and a lastfile  
25 index for the respective terminal frames of each file, the movie further including a  
26 control file containing linking data for linking terminal indices to link segments  
27 together.

28  
29 19) A panorama movie according to claim 18 in which the control file includes  
30 orientation information defining a new direction of travel to proceed from a terminal  
31 frame of first segment along a second segment linked to the first segment.

32  
33 20. A panorama movie according to claim 19 in which a third segments is also  
34 linked to the terminal frame of first segment, the control file orientation information

1 including an alternative new direction of travel selectable by the user during playback  
2 upon reaching the terminal frame of first segment, to branch the movie.

3  
4 21) A panorama movie according to claim 19 in which the new direction of travel  
5 is selectable by the user selecting a viewing direction that approximately coincides  
6 with the new direction of travel.

7  
8 22) A panorama movie method for simulating movement through  
9 multidimensional space comprising the steps of:  
10 capturing a plurality of key-frame images consisting of respective panoramic  
11 imagery taken from a plurality of spatial positions in multidimensional space;  
12 storing said key-frame images in a file in a storage medium;  
13 indexing said key-frame images within a key-frame directory according to a  
14 position of each key-frame within the file and storing said key-frame directory on said  
15 storage medium;  
16 displaying a portion of a first key-frame image according to a user position  
17 and viewing direction within the multidimensional space;  
18 accessing said key-frame directory to determine a next key-frame image to be  
19 displayed according to a user travel speed and travel direction; and  
20 displaying a portion of a second key-frame image subsequent to the display  
21 of said first key-frame image according to a user position and viewing direction.

22  
23 23) The method of claim 22 wherein the step of capturing the plurality of key-  
24 frame images includes taking photographs.

25  
26 24) The method of claim 22 wherein the step of capturing the plurality of key-  
27 frame images includes rendering the images within a computer system.

28  
29 25) The method of claim 22 wherein a forward or reverse travel direction is  
30 determined by the order in which the keyframes are accessed.

31  
32 26) The method of claim 22 wherein a travel speed is determined by accessing  
33 each nth key frame where n is an integer.

1 27) The method of claim 22 wherein a first segment of key-frame images is linked  
2 to a second segment of key-frame images by a control file stored in the storage  
3 medium, the control file including stored travel direction for the second segment  
4 which is selected by the user selecting a viewing direction at a terminal frame of the  
5 first segment that approximately coincides with the stored travel direction for the  
6 second segment.

7  
8 28) The method of claim 22 wherein the user actuates an input device to move  
9 from a first position through a series of subsequent positions, the method including  
10 display successive views of imagery from each of the successive positions.

11  
12 29) The method of claim 22 including changing the viewing direction using the  
13 input device.

14  
15 30) The method of claim 22 including superimposing objects into the display of  
16 multiple successive frames at a location that coincides with apparent movement in  
17 the 3-D space.

18  
19 31) The method of claim 22 including compressing each panorama image using  
20 intraframe compression to form the key-frame images, and during playback selecting  
21 a portion of the key frame that includes the desired viewed selected by the user for  
22 local decompression.

23  
24 32) The method of claim 22 including transforming the panorama images to a  
25 rectangular projection prior to compression.

26  
27 33) The system recited in claim 1 wherein said program adds at least one sound  
28 track to each panorama.

29  
30 34) The system recited in claim 1 wherein said sequence of images includes break  
31 points where said sequence can continue in at least one of two directions.

32  
33 35) The system recited in claim 1 wherein the portion of said linked images displayed  
34 is determined by the direction of view selected by a user and wherein said direction  
35 of view remains constant at break points until changed by a user.

1/12

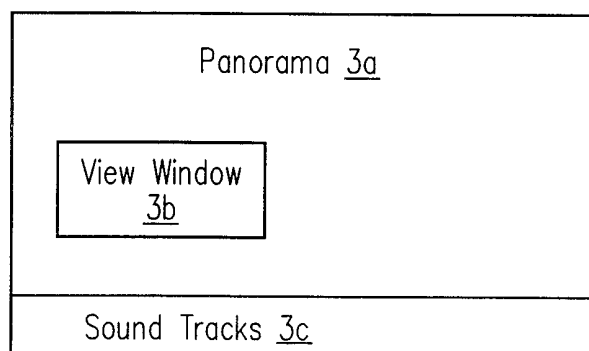


FIG. 1

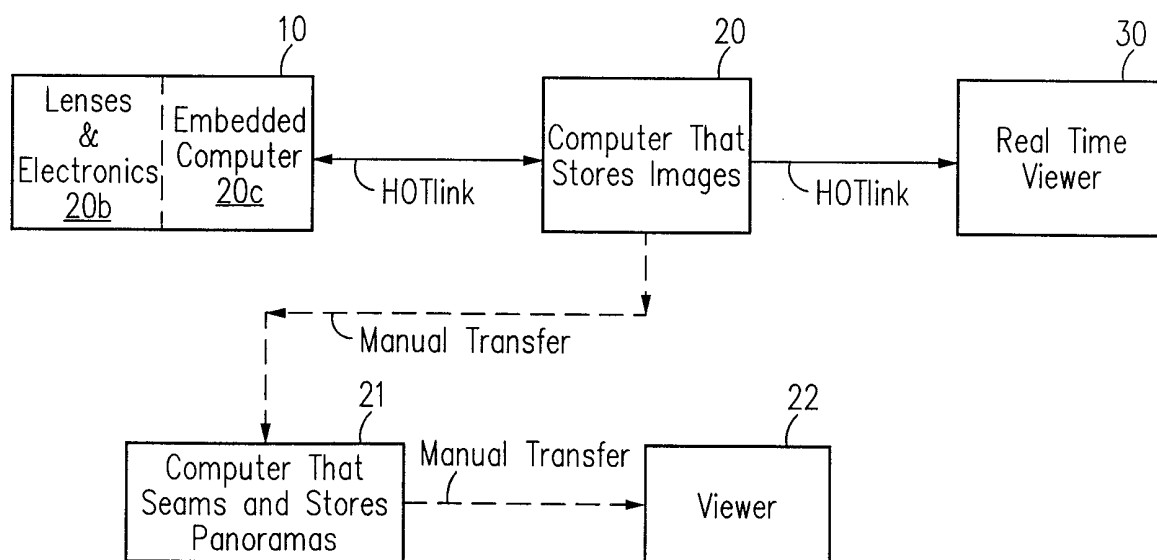


FIG. 2

2/12

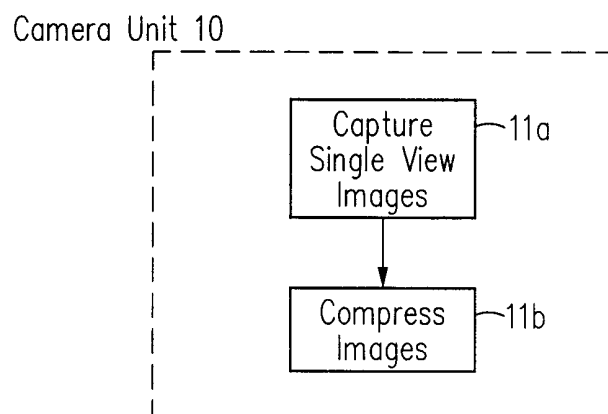


FIG. 3A

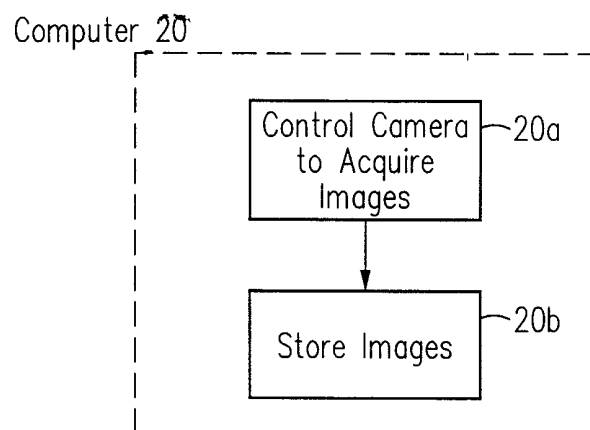


FIG. 3B

3/12

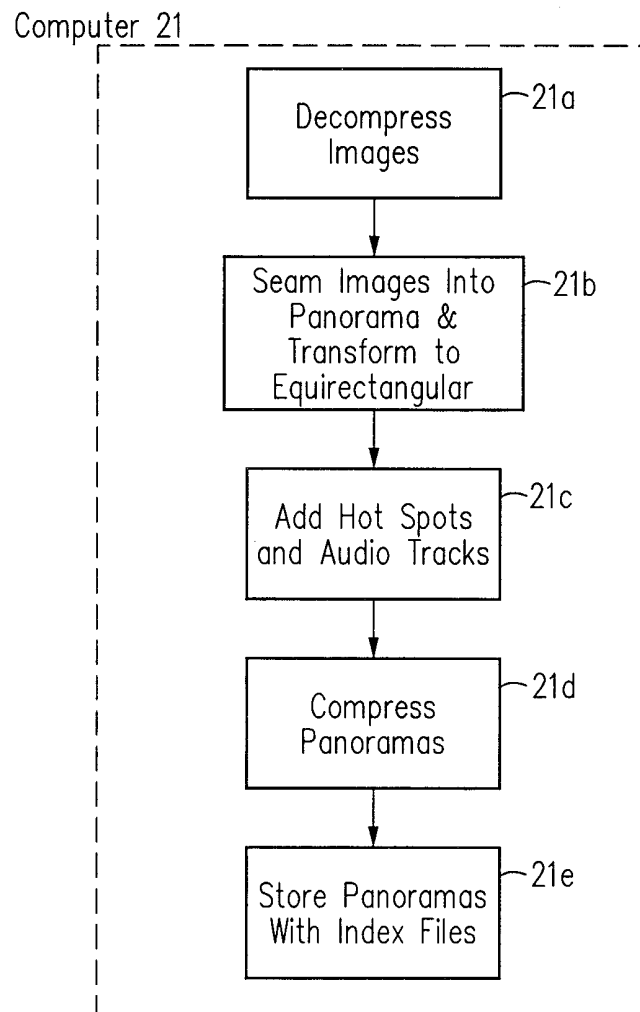


FIG. 3C

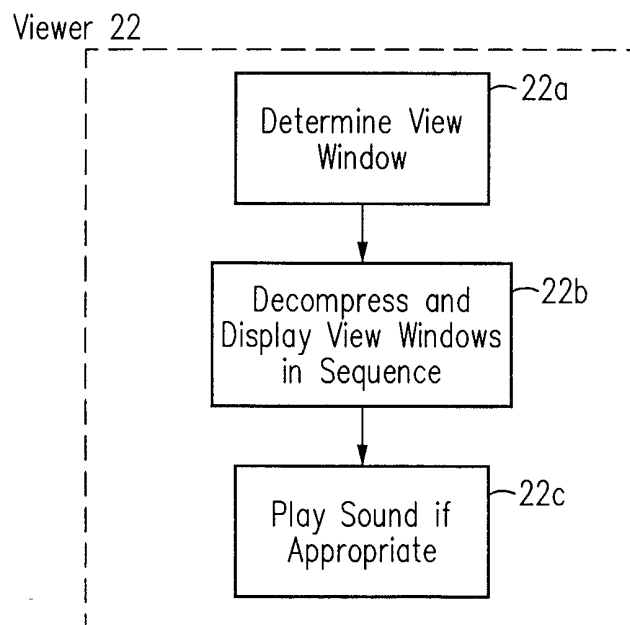


FIG. 3D

4/12

FIG. 4A

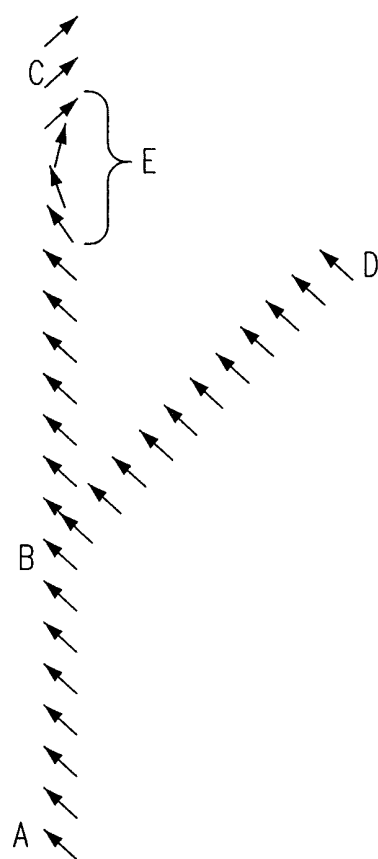
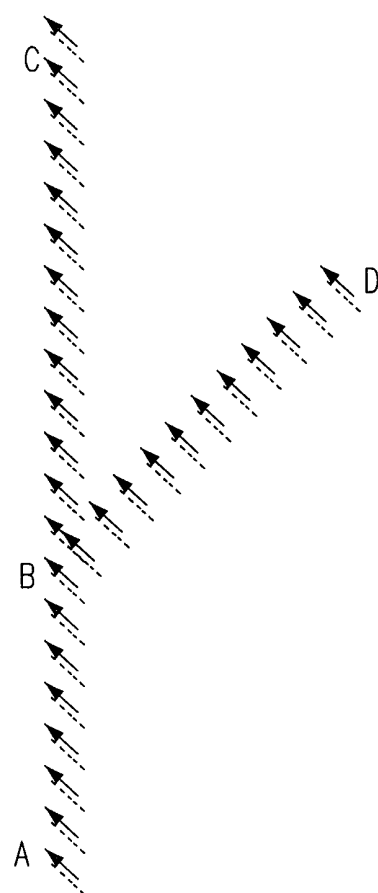


FIG. 4B





5/12

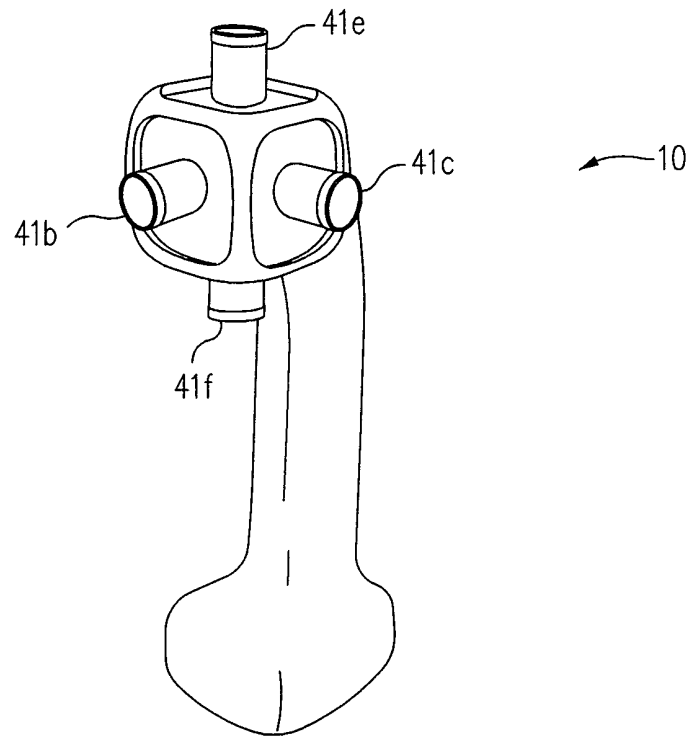


FIG. 5A

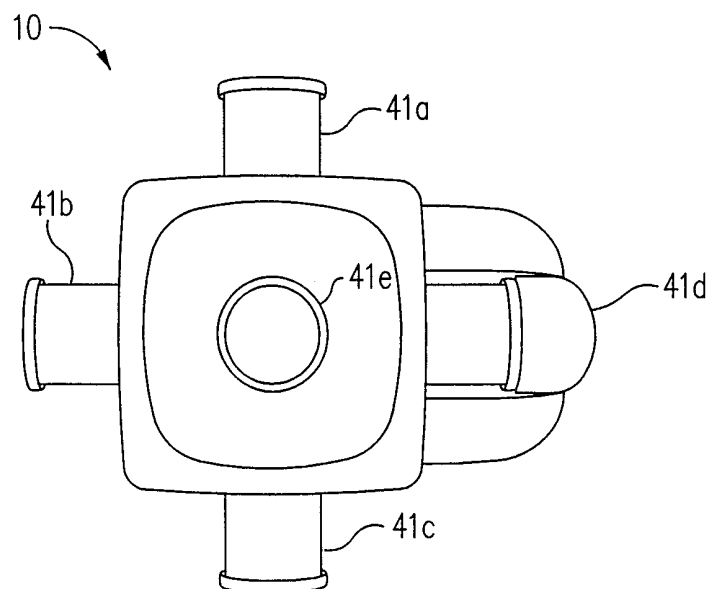


FIG. 5B

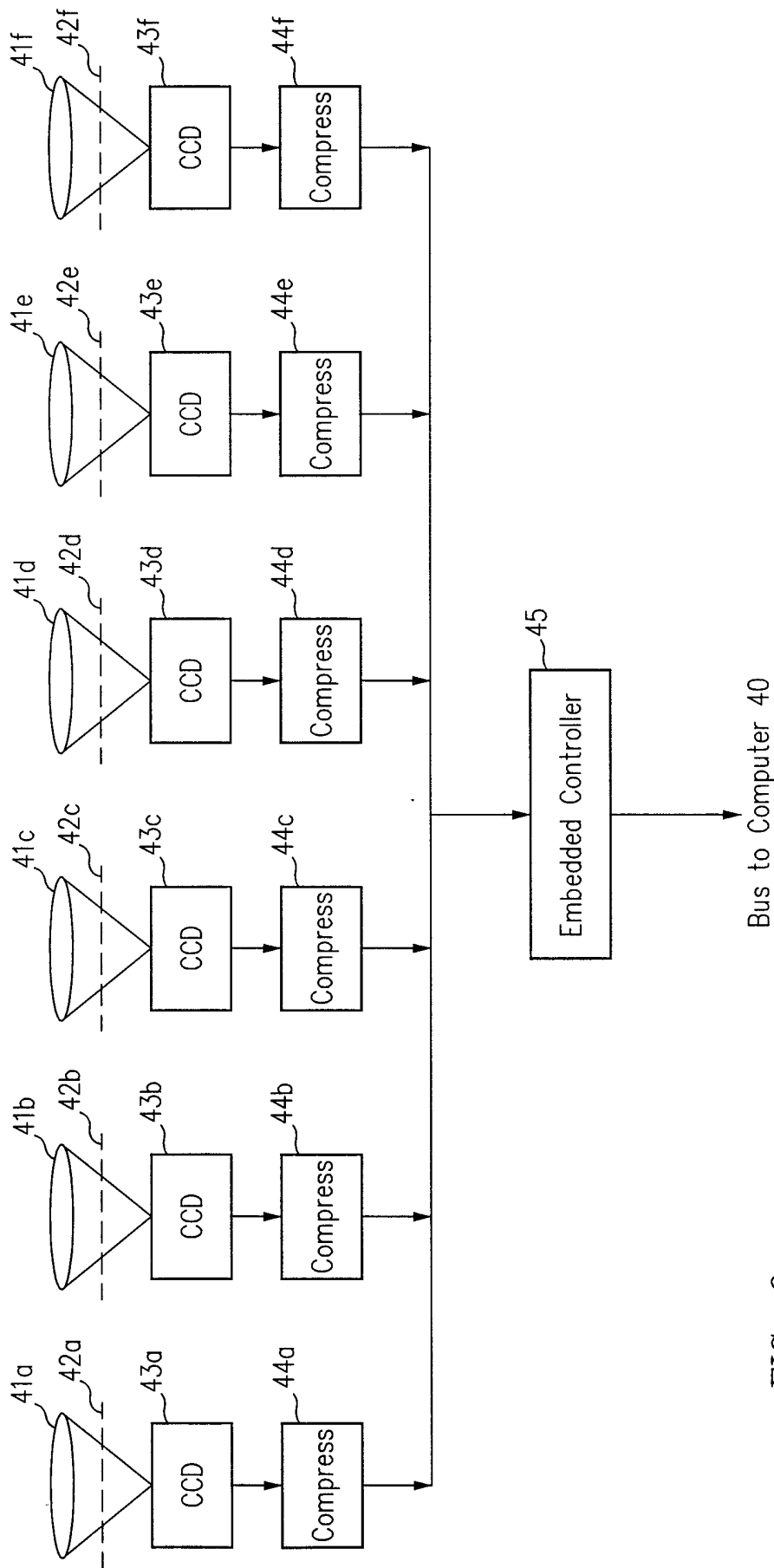


FIG. 6

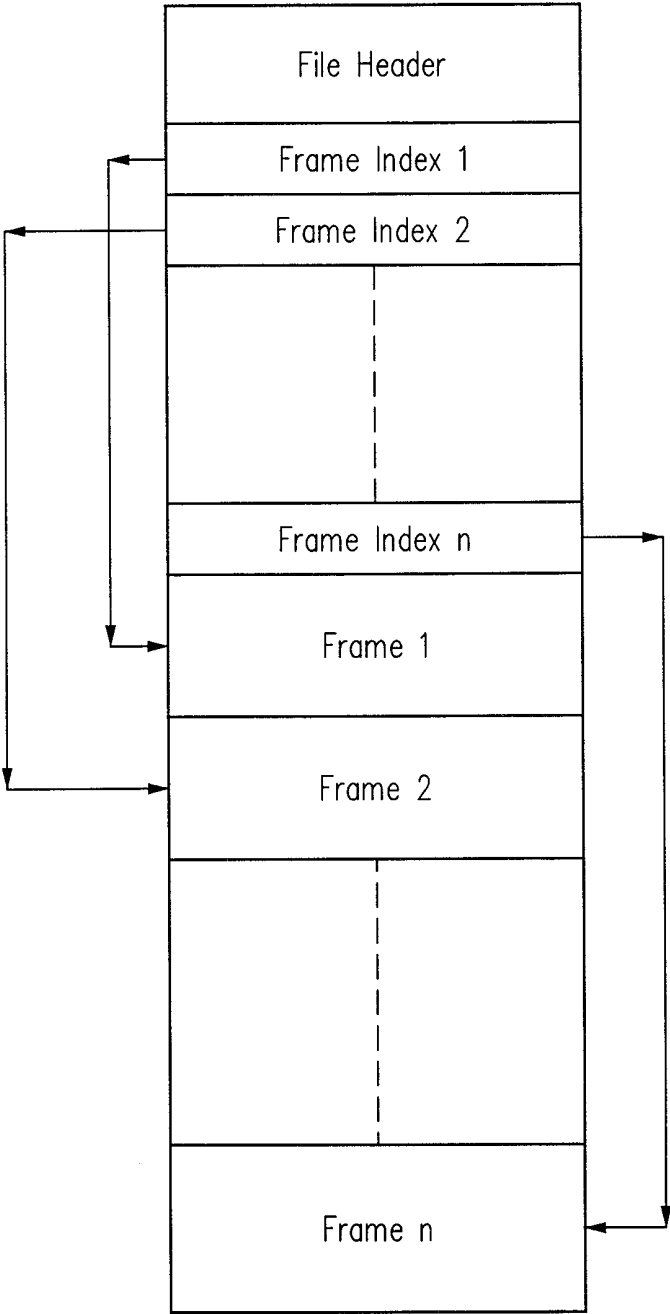


FIG. 7

8/12

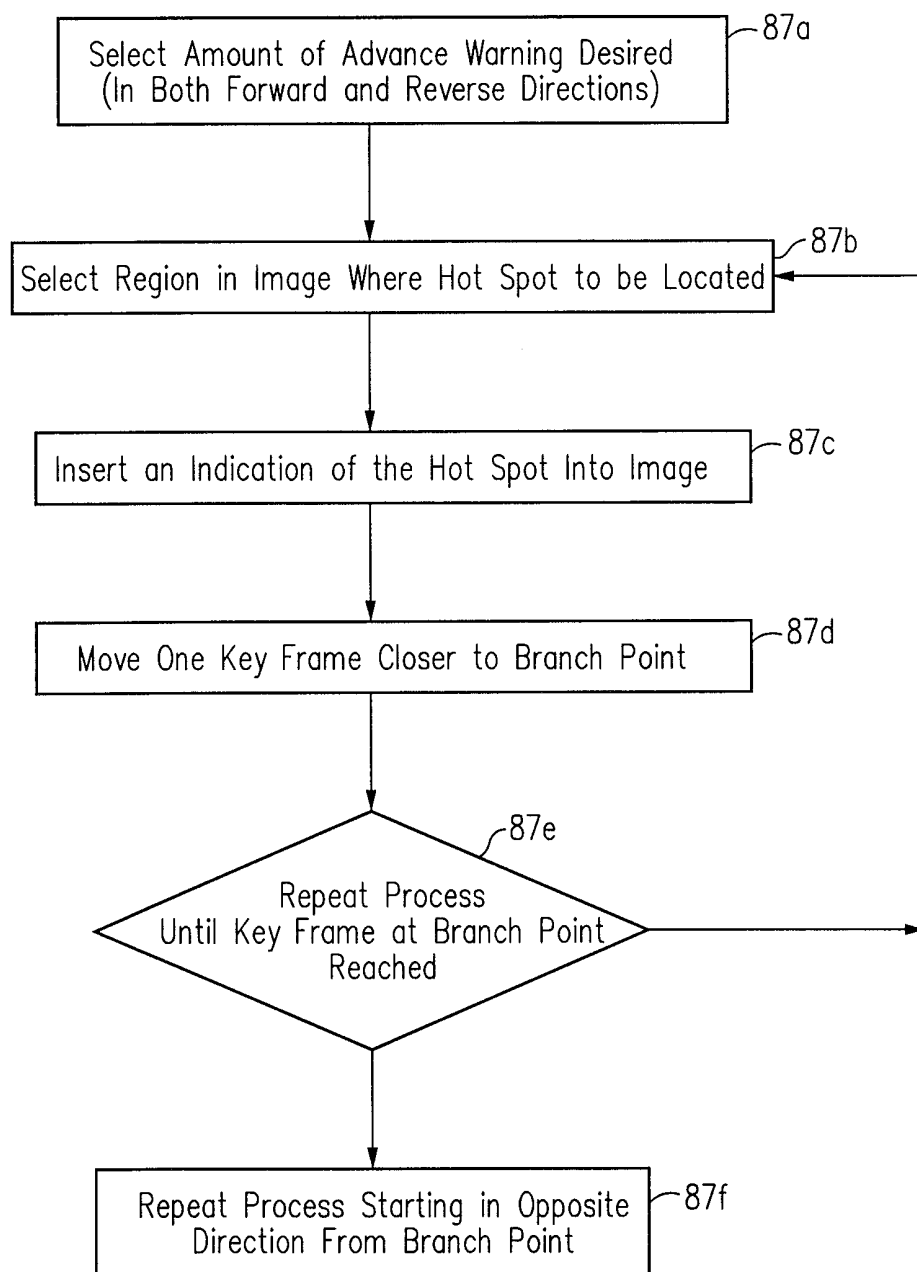


FIG. 8

9/12

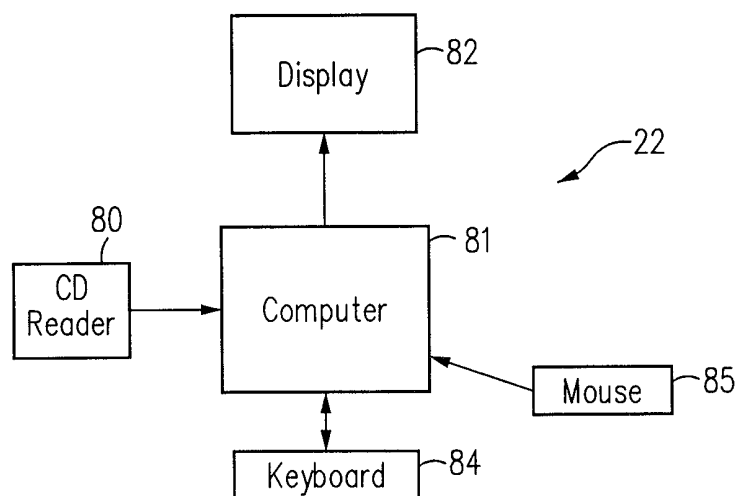


FIG. 9A

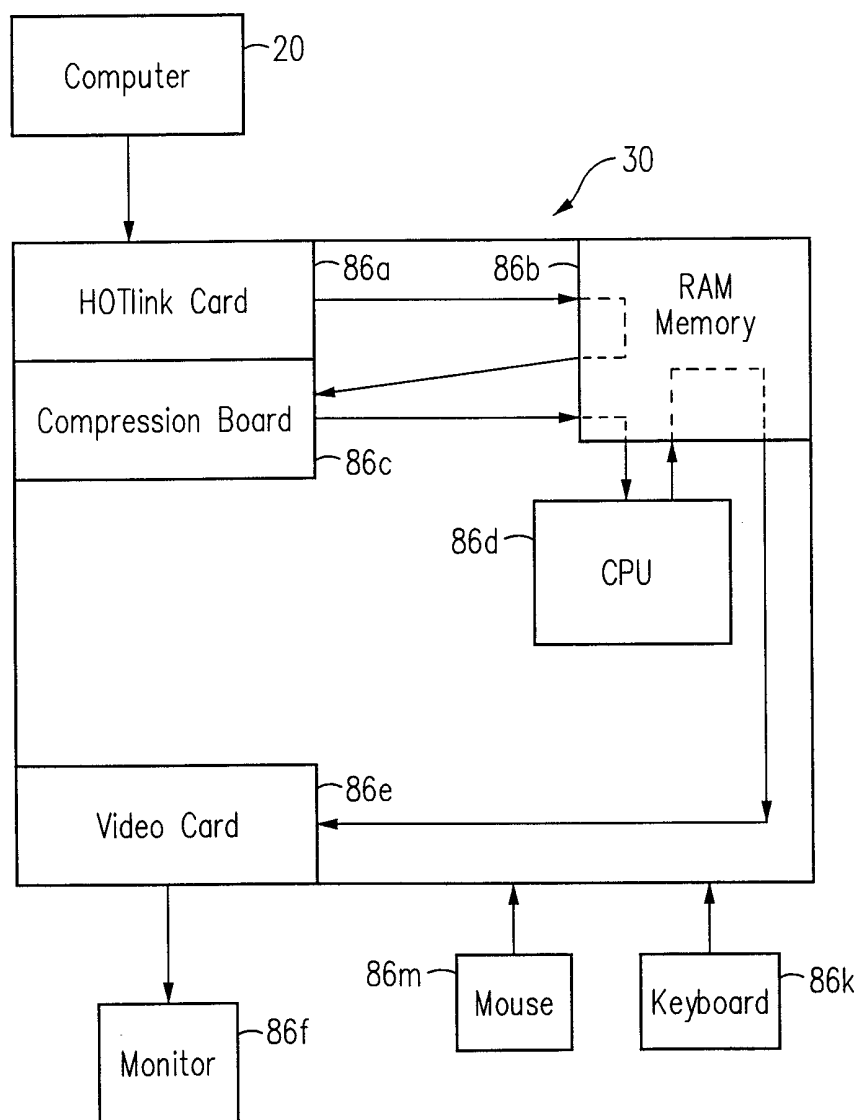


FIG. 9B

10/12

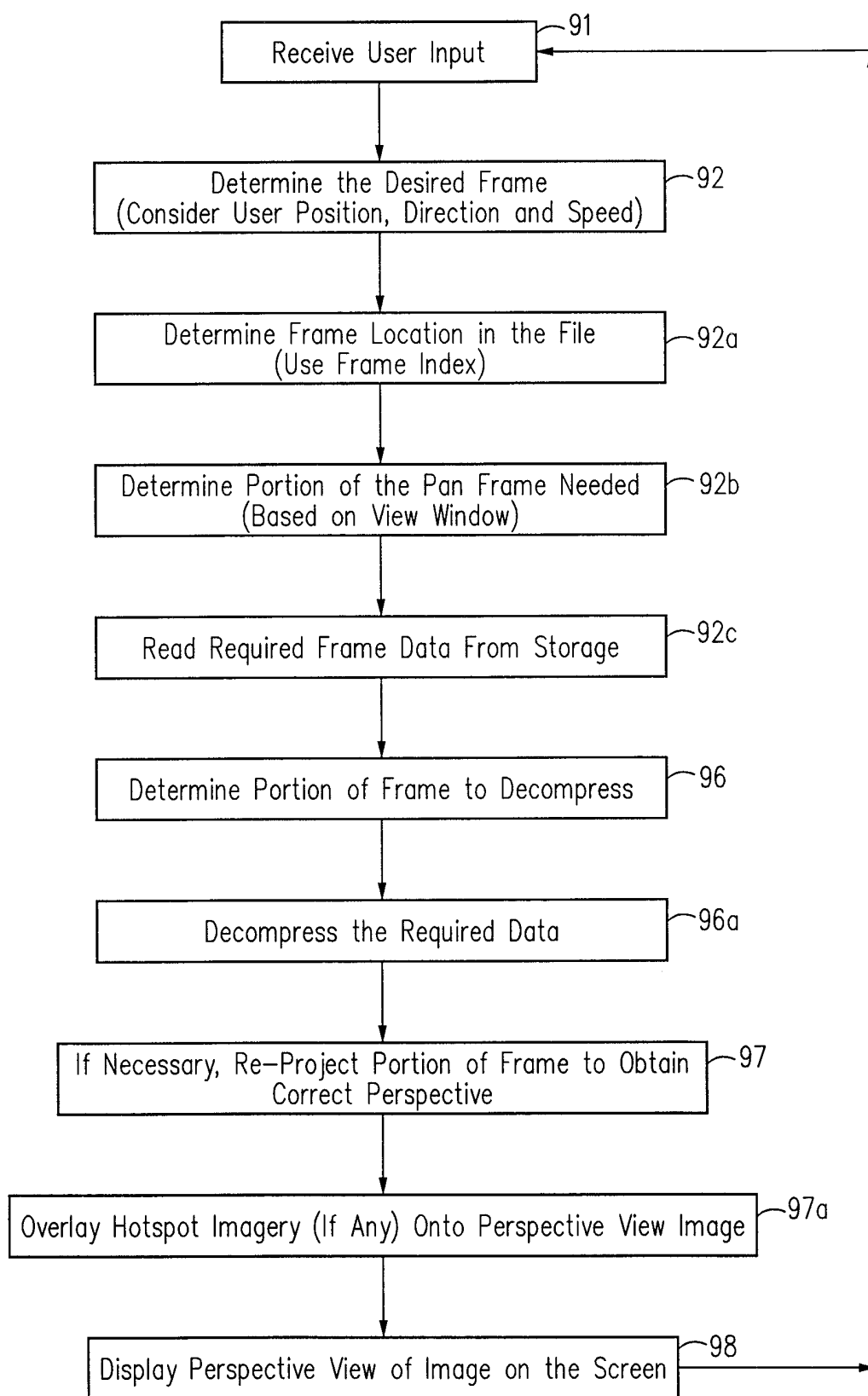


FIG. 10

11/12

Fa	Image Data	Audio A	Audio B	Audio C	Audio D	Audio E	Control Info
Fb	Image Data	Audio A	Audio B	Audio C	Audio D	Audio E	Control Info
Fc	Image Data	Audio A	Audio B	Audio C	Audio D	Audio E	Control Info
Fd	Image Data	Audio A	Audio B	Audio C	Audio D	Audio E	Control Info
Fe	Image Data	Audio A	Audio B	Audio C	Audio D	Audio E	Control Info
Fg	Image Data	Audio A	Audio B	Audio C	Audio D	Audio E	Control Info
Fh	Image Data	Audio A	Audio B	Audio C	Audio D	Audio E	Control Info
Fi	Image Data	Audio A	Audio B	Audio C	Audio D	Audio E	Control Info

FIG. 11

12/12

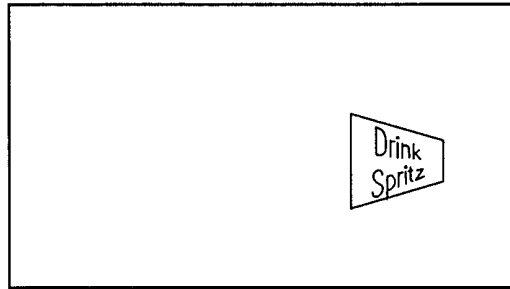


FIG. 12A

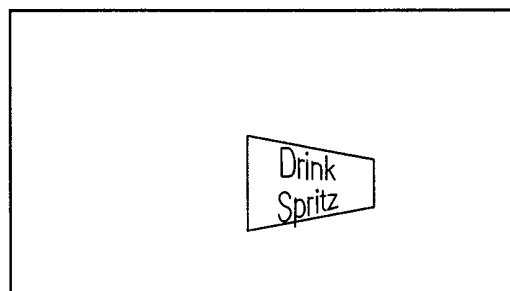


FIG. 12B

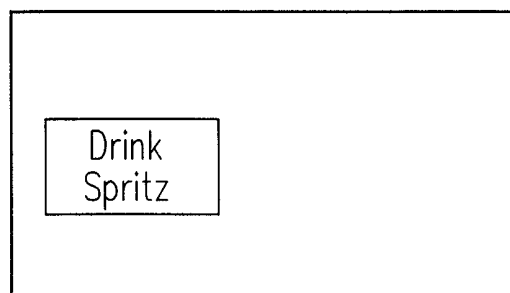


FIG. 12C