US 20040201688A1

(19) United States

(12) Patent Application Publication

(10) Pub. No.: US 2004/0201688 A1

(43) Pub. Date: Oct. 14, 2004

Wolf et al.

(54) ACCESSING IMAGE FILES STORED IN A DIGITAL CAMERA BY A HOST COMPUTER

(75) Inventors: **Edward O. Wolf**, Rochester, NY (US); **Lonne R. Lyon**, Rochester, NY (US)

Correspondence Address:
Thomas H. Close
Patent Legal Staff
Eastman Kodak Company
343 State Street
Rochester, NY 14650-2201 (US)

(73) Assignee: **Eastman Kodak Company**

(21) Appl. No.: **10/100,396**

(22) Filed: **Mar. 18, 2002**

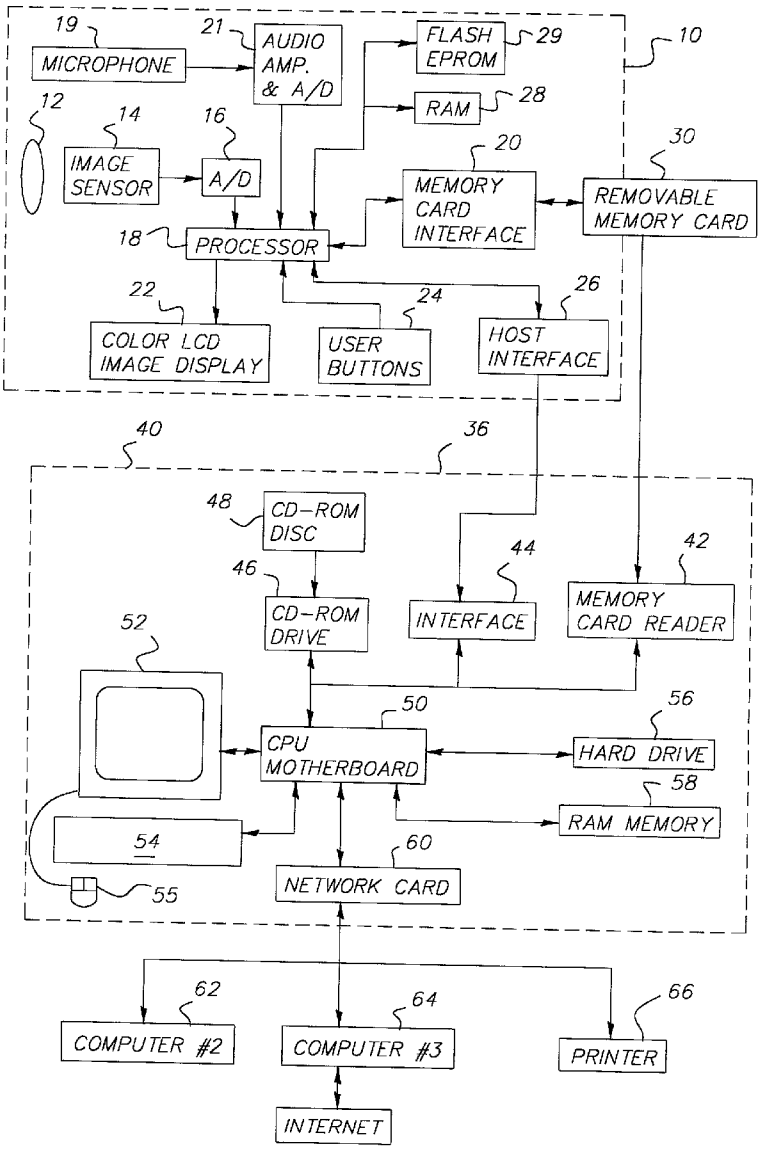
Publication Classification

(51) Int. Cl.[7] H04N 5/225

(52) U.S. Cl. 348/207.1

(57) **ABSTRACT**

A method is disclosed for a host computer to selectively operate on a plurality of image files captured by a digital camera, each image file having at least one digital image. The plurality of captured image files are stored in a memory in the digital camera. The digital camera memory is coupled to the host computer so that the host computer identifies the plurality of captured image files. The host computer identifies the digital camera memory as though it were a file system of an additional hard drive memory for accessing the captured image files. The captured image files are then accessed and selectively transferred from the digital camera memory to the host computer.



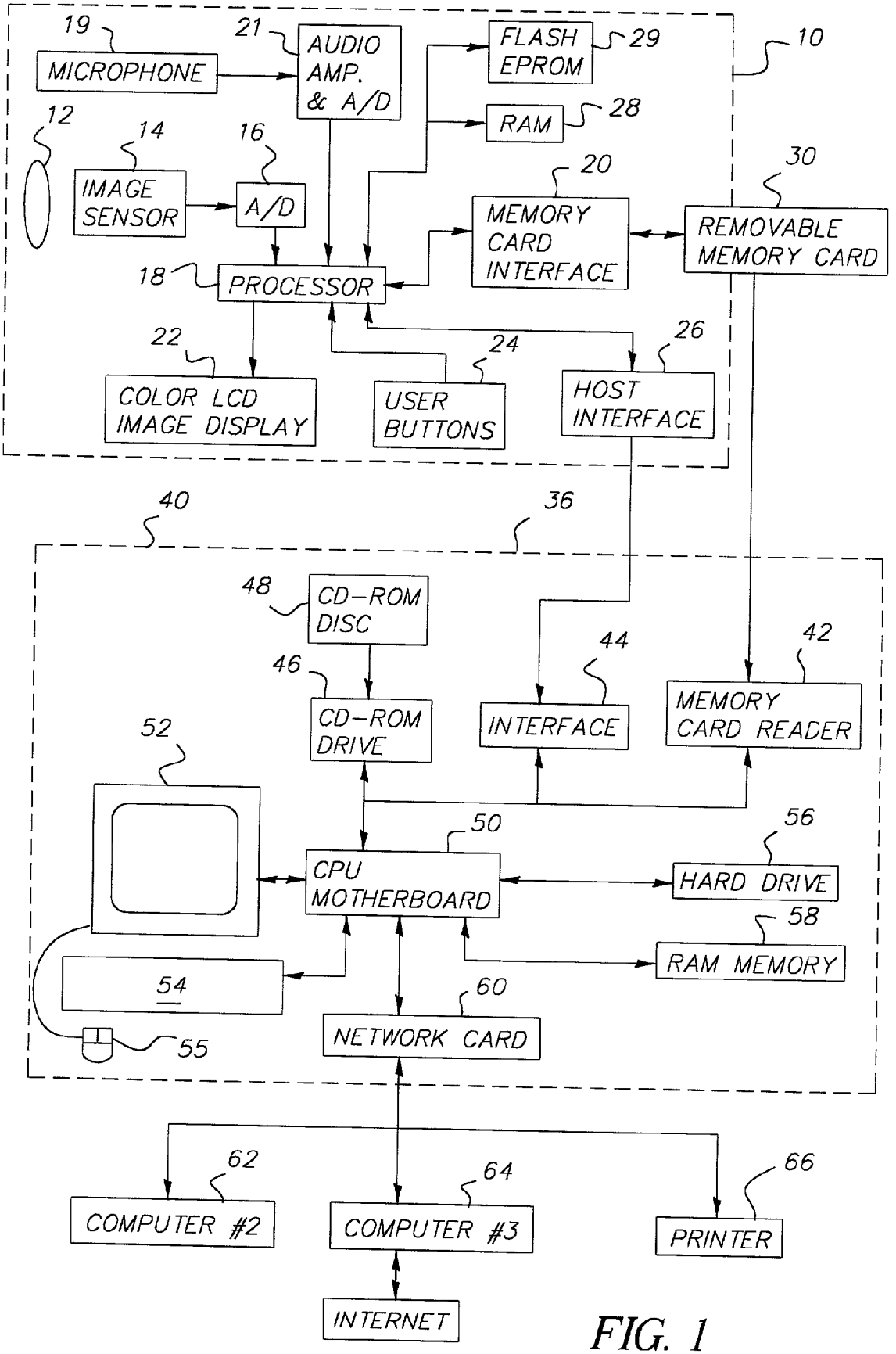

19
MICROPHONE
21
AUDIO AMP. & A/D
FLASH EPROM
29
10
RAM
28
12
14
IMAGE SENSOR
16
A/D
20
MEMORY CARD INTERFACE
30
REMOVABLE MEMORY CARD
18
PROCESSOR
22
COLOR LCD IMAGE DISPLAY
24
USER BUTTONS
26
HOST INTERFACE
40
36
48
CD-ROM DISC
46
CD-ROM DRIVE
44
INTERFACE
42
MEMORY CARD READER
52
50
CPU MOTHERBOARD
56
HARD DRIVE
58
RAM MEMORY
54
60
NETWORK CARD
55
62
COMPUTER #2
64
COMPUTER #3
66
PRINTER
INTERNET

FIG. 1

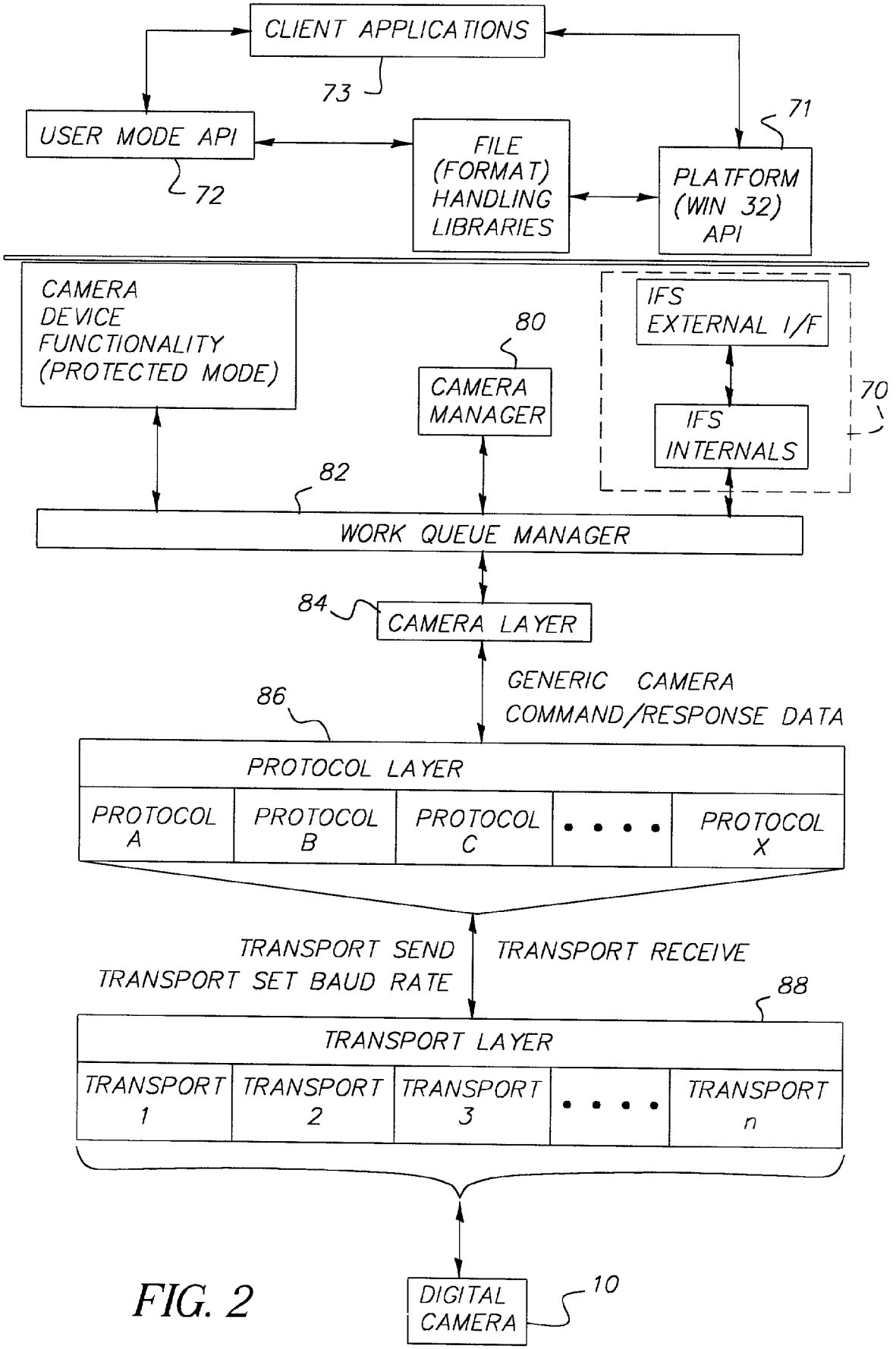

CLIENT APPLICATIONS
73
USER MODE API
72
FILE (FORMAT) HANDLING LIBRARIES
71
PLATFORM (WIN 32) API
CAMERA DEVICE FUNCTIONALITY (PROTECTED MODE)
80
CAMERA MANAGER
IFS EXTERNAL I/F
IFS INTERNALS
70
82
WORK QUEUE MANAGER
84
CAMERA LAYER
GENERIC CAMERA COMMAND/RESPONSE DATA
86
PROTOCOL LAYER
PROTOCOL A
PROTOCOL B
PROTOCOL C
PROTOCOL X
TRANSPORT SEND
TRANSPORT SET BAUD RATE
TRANSPORT RECEIVE
88
TRANSPORT LAYER
TRANSPORT 1
TRANSPORT 2
TRANSPORT 3
TRANSPORT n
DIGITAL CAMERA
10

FIG. 2

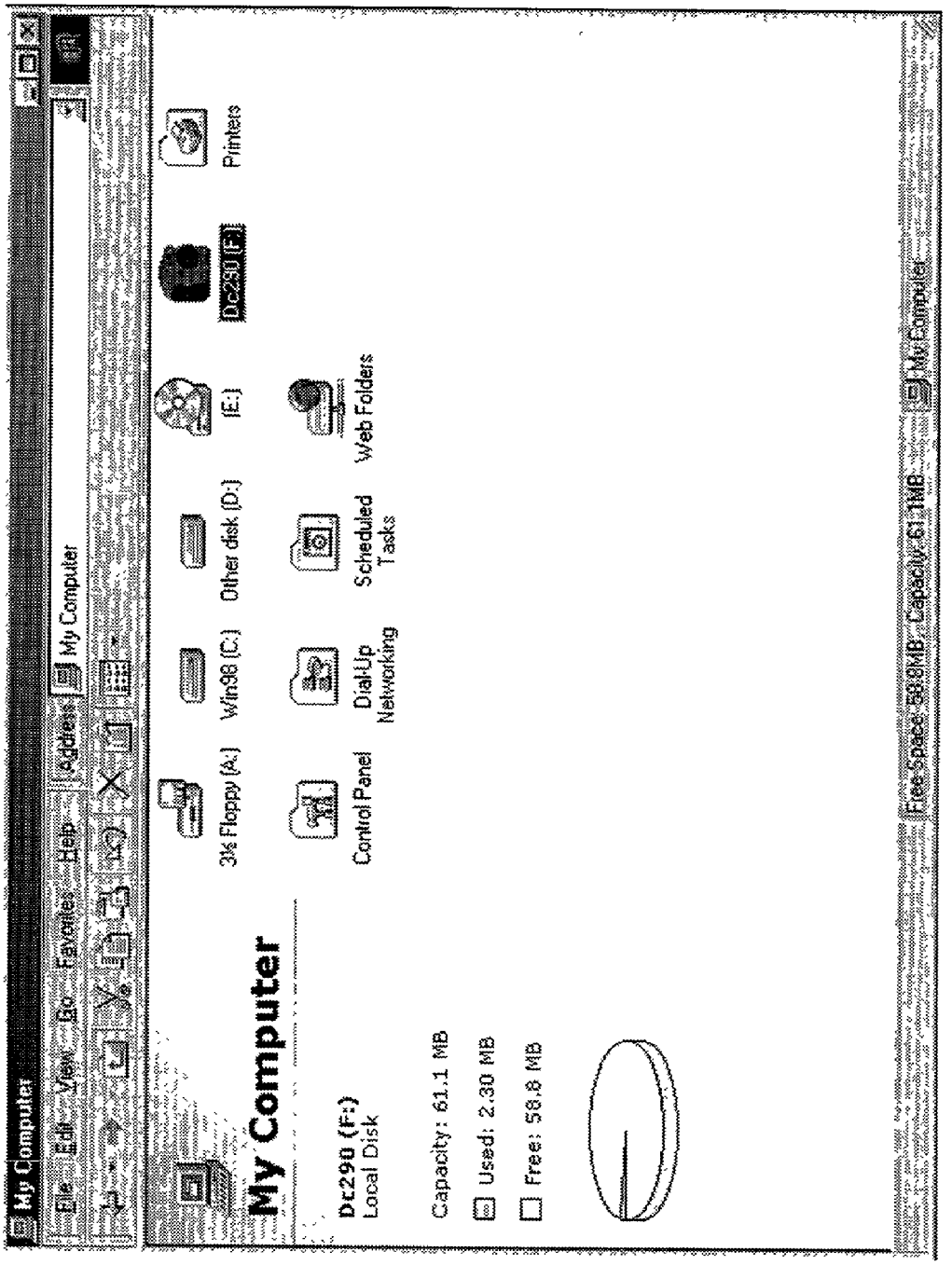

My Computer
File Edit View Go Favorites Help
Address My Computer
My Computer
3½ Floppy (A:)
Win98 (C:)
Other disk (D:)
(E:)
Dc290 (F:)
Printers
Control Panel
Dial-Up Networking
Scheduled Tasks
Web Folders
Dc290 (F:)
Local Disk
Capacity: 61.1 MB
Used: 2.30 MB
Free: 58.8 MB
Free Space: 58.8MB Capacity: 61.1MB
My Computer

Fig. 3A

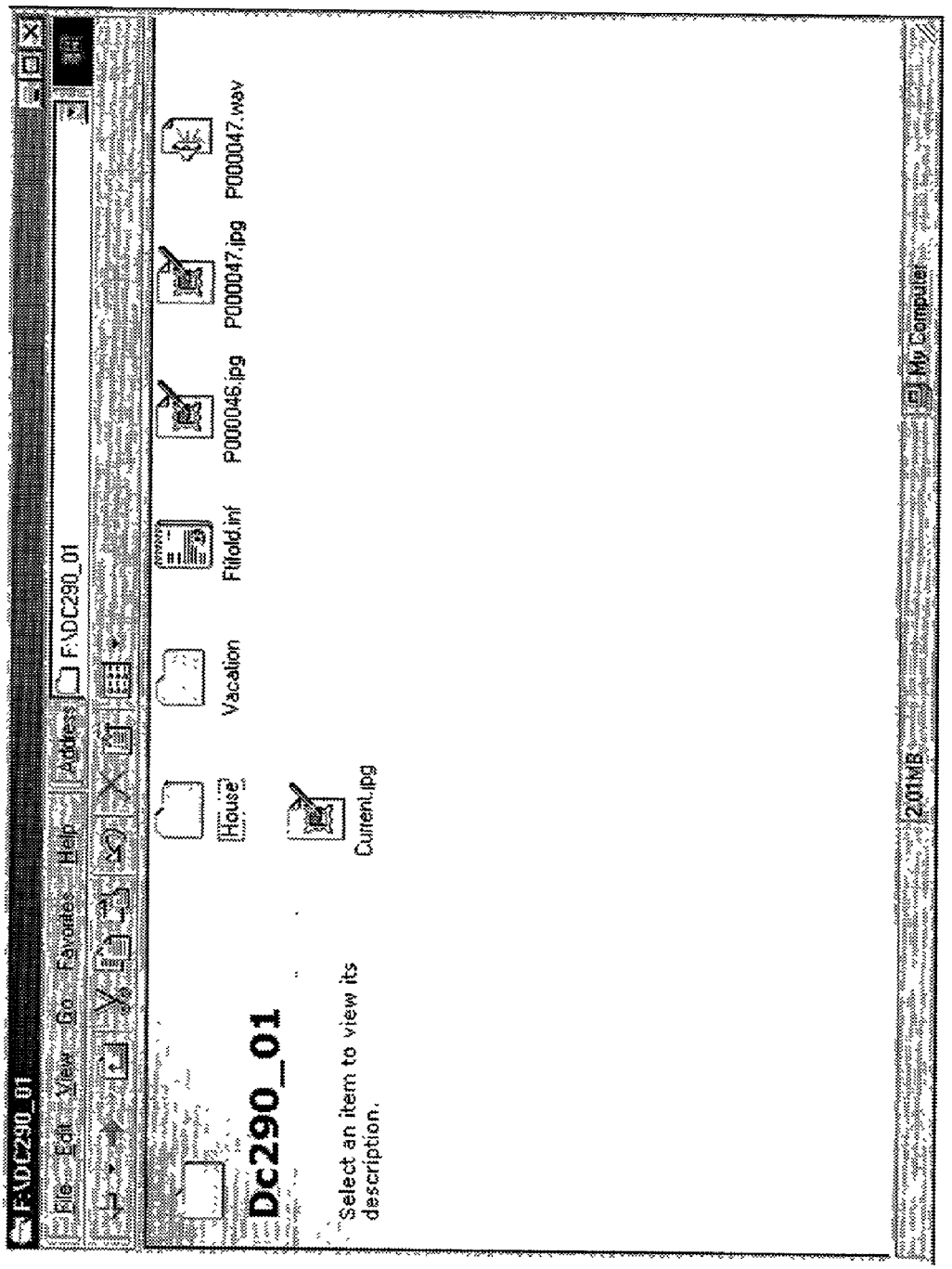

F:\DC290_01
File Edit View Go Favorites Help
Address F:\DC290_01
Dc290_01
Select an item to view its description.
House
Vacation
Current.jpg
Ffifold.inf
P0000046.jpg
P0000047.jpg
P0000047.wav
2.01MB
My Computer

Fig. 3B

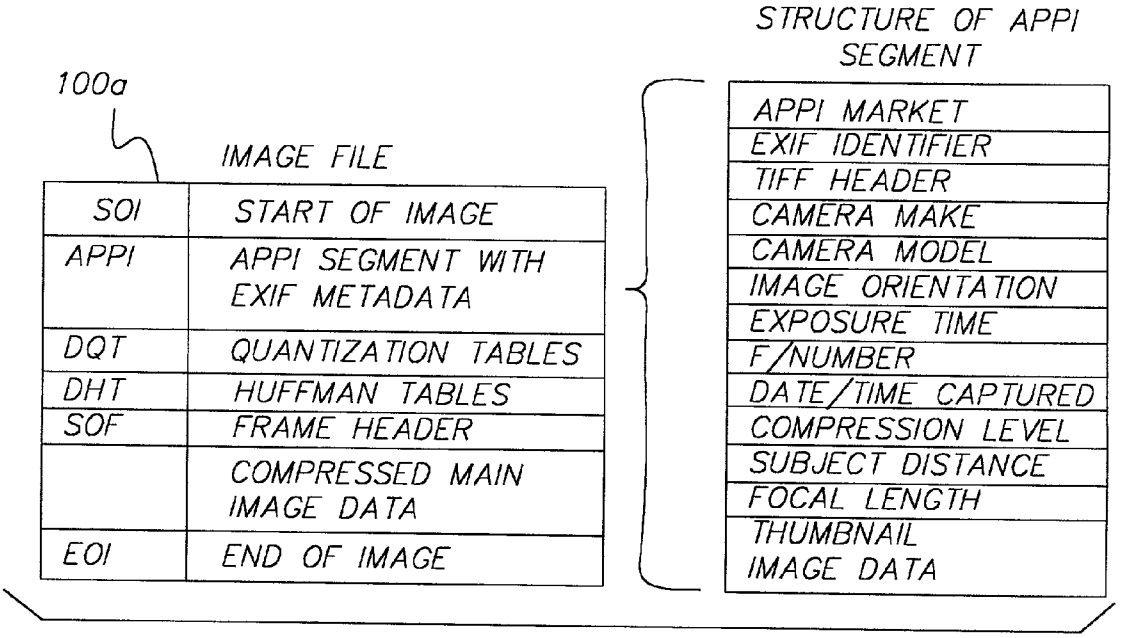

100a
IMAGE FILE
SOI
START OF IMAGE
APPI
APPI SEGMENT WITH EXIF METADATA
DQT
QUANTIZATION TABLES
DHT
HUFFMAN TABLES
SOF
FRAME HEADER
COMPRESSED MAIN IMAGE DATA
EOI
END OF IMAGE
STRUCTURE OF APPI SEGMENT
APPI MARKET
EXIF IDENTIFIER
TIFF HEADER
CAMERA MAKE
CAMERA MODEL
IMAGE ORIENTATION
EXPOSURE TIME
F/NUMBER
DATE/TIME CAPTURED
COMPRESSION LEVEL
SUBJECT DISTANCE
FOCAL LENGTH
THUMBNAIL IMAGE DATA

FIG. 4A

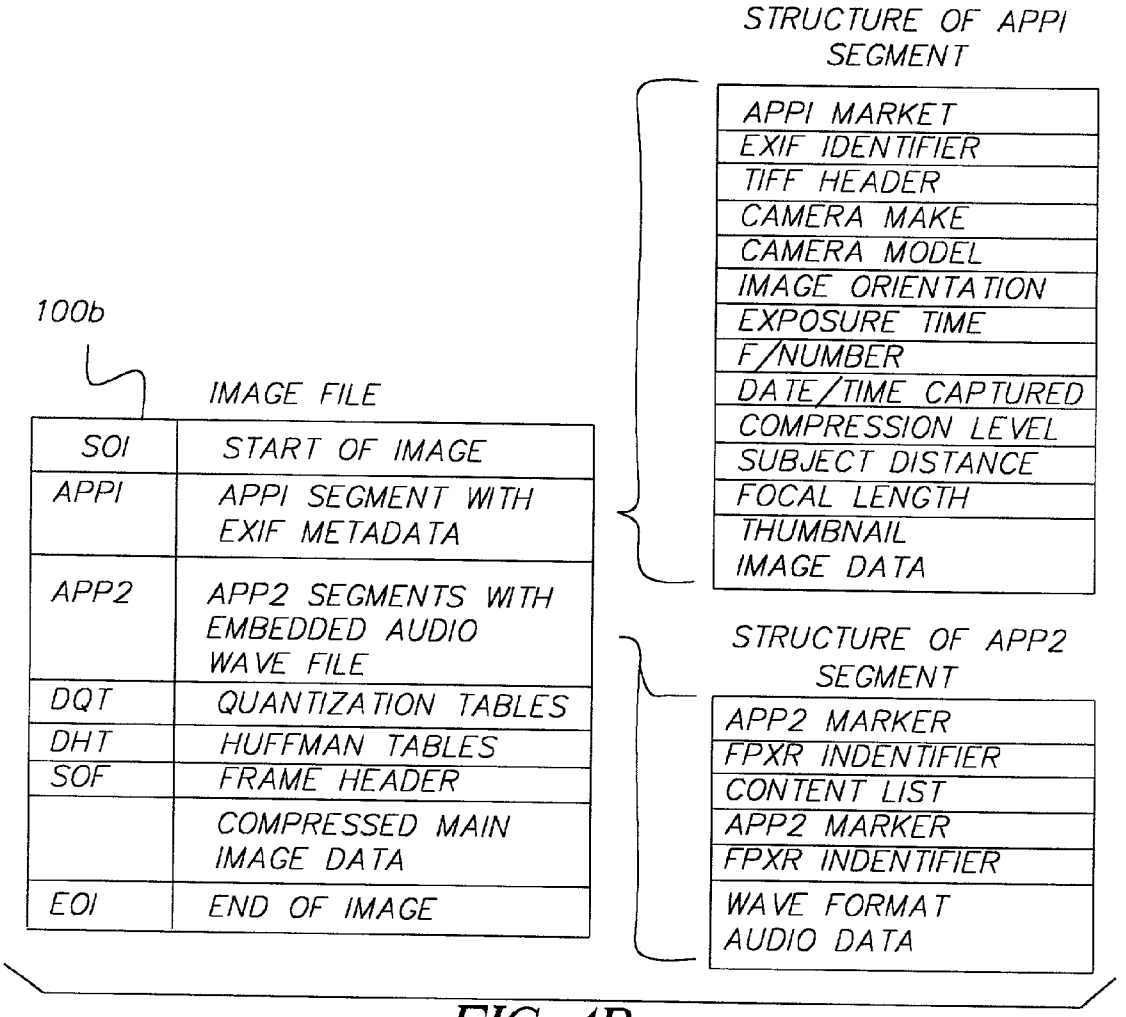

100b
IMAGE FILE
SOI
START OF IMAGE
APPI
APPI SEGMENT WITH EXIF METADATA
APP2
APP2 SEGMENTS WITH EMBEDDED AUDIO WAVE FILE
DQT
QUANTIZATION TABLES
DHT
HUFFMAN TABLES
SOF
FRAME HEADER
COMPRESSED MAIN IMAGE DATA
EOI
END OF IMAGE
STRUCTURE OF APPI SEGMENT
APPI MARKET
EXIF IDENTIFIER
TIFF HEADER
CAMERA MAKE
CAMERA MODEL
IMAGE ORIENTATION
EXPOSURE TIME
F/NUMBER
DATE/TIME CAPTURED
COMPRESSION LEVEL
SUBJECT DISTANCE
FOCAL LENGTH
THUMBNAIL IMAGE DATA
STRUCTURE OF APP2 SEGMENT
APP2 MARKER
FPXR INDENTIFIER
CONTENT LIST
APP2 MARKER
FPXR INDENTIFIER
WAVE FORMAT AUDIO DATA

FIG. 4B

ACCESSING IMAGE FILES STORED IN A DIGITAL CAMERA BY A HOST COMPUTER

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] Reference is made to commonly-assigned U.S. patent application Ser. No. 09/781,594, filed Oct. 12, 2000 entitled "Producing Icons For Accessing Image Files Transferred From A Digital Camera" by Edward Wolf et al., and commonly assigned U.S. Provisional Patent Application Serial No. 60/159,162 (docket 81,770), filed Oct. 13, 1999, entitled "Prioritizing The Transfer of Image Files from a Digital Camera to a Host Computer" by Edward Wolf et al.

FIELD OF THE INVENTION

[0002] This invention relates to accessing images stored in a digital camera by a host computer.

BACKGROUND OF THE INVENTION

[0003] Digital images are often produced by electronic still cameras, such as the Kodak Digital Science DC265™ camera sold by Eastman Kodak Company. Exemplary digital cameras are described in more detail in commonly assigned U.S. Pat. Nos. 5,828,406, 5,633,678, and 5,477,264, the disclosures of which are incorporated herein by reference. These cameras include an image sensor, an analog to digital converter, and a storage device for storing the digital image files. There are many storage devices on which these digital images can be stored, including floppy magnetic discs, hard magnetic disc drives, and solid state memory (e.g., flash memory) cards.

[0004] The images can be download by removing the memory card (e.g., CompactFlash card) from the digital camera and inserting it into a card reader attached to a host computer, or by connecting the digital camera and host computer together via a cable (e.g., Universal Serial Bus) or wireless (e.g., IrDA) interface. Software provided with the digital camera is typically installed on the host computer and used to control the camera interface. This software typically provides commands that allow thumbnail (i.e., reduced resolution) images and full size images to be transferred from the camera to the host computer.

[0005] One type of design for implementing such software is described in U.S. Pat. No. 5,848,420, the disclosure of which is herein incorporated by reference.

[0006] Typically, there is a problem when a user transfers image files from a digital camera to a host computer. When the user selects an image to view, the images have to be transferred from the digital camera to the host computer following the user selection. A problem associated with image transfer is that each separate time that a user requests that an image file be accessed or opened, the entire image file is transferred from the digital camera to the host computer without reference to whether the image file had previously been transferred. Since image files contain not only a digital image, but also other components, if the user is only interested in the components, the user must access the entire image file.

SUMMARY OF THE INVENTION

[0007] It is an object of the present invention to provide a way for a host computer to simply and easily access images from a digital camera.

[0008] It is another object of the present invention to permit access to previously transferred image files or portions thereof without retransfer of the image file from the digital camera to the host computer.

[0009] It is a further object of the present invention to permit a user to selectively access portions of an image file.

[0010] These objects are achieved by a method for a host computer to selectively operate on a plurality of image files captured by a digital camera, each image file having at least one digital image, the method comprising the steps of:

[0011] (a) storing the plurality of captured image files in a memory in the digital camera;

[0012] (b) coupling the digital camera memory to the host computer so that the host computer identifies the plurality of captured image files;

[0013] (c) the host computer identifying the digital camera memory as though it were a file system of an additional hard drive memory for accessing the captured image files; and

[0014] (d) accessing and selectively transferring the captured image files from the digital camera memory to the host computer.

[0015] The invention provides a software driver on the host computer that treats the digital camera as a connection dependent file system for peripheral storage devices, for example, as an "installable file system" (IFS) for Windows 95 and Windows 98 Operating System (OS) based personal computers (PCs). This IFS appears to a user as an additional drive. Images appear as files, and albums appear as directories or folders.

BRIEF DESCRIPTION OF THE DRAWINGS

[0016] **FIG. 1** illustrates a digital imaging system including a digital camera and a host computer that implements the present invention;

[0017] **FIG. 2** is a block diagram of software components in accordance with the present invention;

[0018] **FIG. 3A** illustrates an exemplary view of a screen on a display monitor of the host computer of **FIG. 1** after the digital camera is connected to the host computer;

[0019] **FIG. 3B** illustrates an additional exemplary view of a screen on a display monitor of the host computer depicting icons which represent files (including image data and audio segment data) and directories (or albums);

[0020] **FIG. 4A** illustrates the image file structure when no audio has been recorded with the image (for example, file P0000046.jpg in FIGS. **3**B); and

[0021] **FIG. 4B** illustrates the image file structure when audio has been recorded with the image (for example, file P0000047.jpg in **FIGS. 3B**).

DETAILED DESCRIPTION OF THE INVENTION

[0022] **FIG. 1** illustrates a system which can be used for controlling a digital camera **10** attached to a host computer **40** using software contained on a compact disc (CD) **48** or other media in accordance with the present invention. The

compact disc **48**, containing the software that implements the methods described in this invention, is inserted into a well-known CD-ROM drive **46** in the host computer **40**. Alternatively, the software can be stored on a floppy magnetic disc (not shown), a removable memory card **30**, or another type of media. The compact disc **48**, floppy disc, or removable memory card **30**, or an alternative type of digital storage media, is supplied to the user along with the digital camera **10**.

[**0023**] The digital camera **10** produces digital images that are stored on the removable memory card **30**. The digital camera **10** includes a lens **12** having an adjustable aperture and shutter (not shown) for focusing light from a scene (not shown) on an image sensor **14**, for example, a single-chip color charge coupled device (CCD), using the well-known Bayer color filter pattern. The analog output signal from the image sensor **14** is converted to digital data by an analog-to-digital (A/D) converter **16**. The digital data is processed by a processor **18**, and the processed digital image file is provided to a memory card interface **20** which stores the digital image file on the removable memory card **30**. Removable memory cards **30** are known to those skilled in the art. For example, the removable memory card **30** can include memory cards adapted to the PCMCIA card interface standard, as described in the *PC Card Standard, Release* 2.0, published by the Personal Computer Memory Card International Association, Sunnyvale, Calif., September 1991. The removable memory card **30** can also be adapted to the Compact Flash interface standard, such as described in the *CompactFlash Specification Version* 1.3, published by the CompactFlash Association, Palo Alto, Calif., Aug. 5, 1998.

[**0024**] The processor **18** performs color interpolation followed by color and tone correction, in order to produce rendered sRGB image data. The rendered sRGB image data is then JPEG compressed and stored as an Exif version 2.1 file on the removable memory card **30**. The Exif image format is defined in "Digital Still Camera Image File Format Standard, Exchangeable image file format for Digital Still Camera: Exif," JEIDA-49-1998, June 1998 by the Japan Electronics Industries Development Association (JEIDA) and the sRGB color space is described in "A standard default color space for the internet—sRGB" by Michael Stokes, et al., available at http://www.color.org/sRGB.html. The processor **18** also provides "thumbnail" size image data to an image display **22**, such as a color liquid crystal display (LCD), which displays the captured image for the user to review. The digital camera **10** is controlled by a series of user buttons **24**.

[**0025**] The digital camera **10** also includes a microphone **19** and an audio amplifier and A/D converter **21**. After capturing an image, the user may press one of the user buttons **24** in order to record audio, for example, annotation by the photographer. The audio signals from the microphone **19** are amplified and converted to digital data by the audio amplifier and A/D converter **21**. The audio signals may be compressed, for example, using the well-known IMA ADPCM compression algorithm, and stored as Flashpix Extension data within the Exif 2.1 image file. The Flashpix audio extension is defined in "Extension to Flashpix version 1.0, Embedded Audio Annotations", Jan. 26, 1998 by the Digital Imaging Group, which is available at http://www.digitalimaging.org/.

[**0026**] The user buttons **24** also allow other camera features and operating modes to be selected. These modes can also be selected via the host computer **40**. The features and modes include self-timer mode, flash mode, focus mode, exposure mode, white balance mode, picture quality (compression level) mode, resolution mode, sleep and power off times, quickview mode, video output format, and zoom position.

[**0027**] After a series of images have been captured by the digital camera **10** and stored on the removable memory card **30**, the removable memory card **30** can be inserted into a memory card reader **42** in the host computer **40**. Alternatively, an interface cable **36** can be used to connect between a host interface **26** in the digital camera **10** and a camera interface **44** in the host computer **40**. The interface cable **36** can conform to, for example, the well-known universal serial bus (USB) interface specification.

[**0028**] A central processing unit (CPU) **50** will be understood to use software in accordance with the present invention that will be described in more detail with reference to FIGS. **2**, **3A-3B** and **4A-4B**. The CPU **50** is coupled to a display monitor **52** and a keyboard **54**. A mouse **55** permits the user to readily communicate with the CPU **50**. The CPU **50** is also connected to a hard drive **56** and to random access memory (RAM) **58**. The CPU **50** also communicates with networked devices via a network card **60**. The network card **60** is connected to a second computer **62**, a third computer **64** which serves as an internet service provider connection to the internet, and to a hardcopy printer **66**.

[**0029**] **FIG. 2** depicts a set of software components in accordance with the present invention for operation on an operating system, such as Windows 95 and Windows 98 OS based PCs, which enable applications to communicate with one or more digital cameras. One of these software components, the installable file system (IFS) **70**, is a connection dependent file system which enables applications to communicate with the digital camera **10** through a Win32 Application Programmers Interface (API) **71** as a file system of a hard drive memory. See, for example, Chapter 16 of "Systems Programming for Windows 95," published by Microsoft Press, Redmond, Wash., Copyright 1996 by Walter Oney. Other components enable applications to access and set the camera properties, as well as initiate a capture of a new image by the digital camera **10**.

[**0030**] A user mode Application Programming Interface (API) **72** controls camera device operation, as well as accessing information recorded in the image file. Client applications **73** access the user mode API **72** through a set of component object model (COM) interfaces (not shown) that enable:

[**0031**] Detection/enumeration of cameras currently attached to the system;

[**0032**] Communication with a specific camera (including registration for camera-related events); and

[**0033**] Accessing data associated with an image file (e.g., thumbnails, and audio segment data).

[**0034**] An image file is accessed through the Win32 API **71**, which then accesses the Installable File System (IFS) **70**. Whenever the disclosure refers to accessing an image file, it

will also be understood that a portion of an image file, such as an audio data segment, can be individually accessed, transferred, and operated on.

[0035] The user mode API **72** uses COM based interfaces. These interfaces are responsible for accessing image file data including any associated data, such as audio segment data, data and time of image capture, originating device, and the like. **FIGS. 4A and 4B** show exemplary image files and their structural organizations. The image file in **FIG. 4A** is numbered **100***a*, and the image file in **FIG. 4B** is numbered **100***b*. The nomenclature used to describe the image files **100***a* and **100***b* use acronyms which are well understood to those skilled in the art. The user mode API **72** also enables a user to control the operation of the digital camera **10**. The operation of the digital camera **10** can be controlled via a network. More particularly, the host computer **40** (shown in **FIG. 1**) selectively operates on a plurality of image files captured by the digital camera **10**. Each image file includes at least one digital image. Typically the image file includes audio segment data. Under the control of the host computer **40** or, alternatively, under the control of a user of the digital camera **10**, a plurality of captured image files is stored in the removable memory card **30** in the digital camera **10**. The digital camera **10** is connected to the host computer **40**, and the host computer **40** identifies the plurality of captured image files stored on the removable memory card **30**. In accordance with the present invention, the host computer **40** identifies the removable memory card **30** as though it were a file system of an additional hard drive memory for accessing the captured image files. The host computer **40** accesses and selectively transfers the captured image files from the removable memory card **30** to the RAM memory **58** of the host computer **40**.

[0036] The Installable File System (IFS) software component **70** enables the operating system to view the digital camera **10** as a file system of an additional hard drive memory. The IFS **70** can be considered to have two components. The first is an external component (IFS-E) which satisfies operating system requests, such as notifying the operating system (OS) of new files and obtaining and relinquishing drive letters from the OS. The second is an internal component (IFS-I) which manages tracking and caching of files. IFS-I also provides convenient storage and flexible information retrieval to other layers.

[0037] A Camera Manager **80** software component is responsible for enabling several of the other components to work together. When the Camera Manager **80** receives a "Camera Arrival" event to identify that the digital camera **10** has been connected to the host computer **40**, it notifies the IFS **70** to obtain a drive identification alphabetic letter (i.e., 3½ floppy drive (A:) shown in **FIG. 3A**) from the system and then populate the drive with file and folder information. The Camera Manager **80** maintains camera instance information, such as the link between the unique identification assigned by a Camera Layer **84** software component and the drive identification alphabetic letter that the IFS **70** obtains from the system. When the Camera Manager **80** receives a "Camera Departure" event to indicate that the digital camera **10** has been disconnected from the host computer **40**, it notifies the IFS **70** to remove the drive that was linked to the disconnected digital camera **10**. The Camera Manager **80** is also responsible for determining what to do when the digital camera **10** is disconnected while an operation is in progress (or work is queued), and then reconnected at a later time.

[0038] A Work Queue Manager **82** software component facilitates image files captured by the digital camera **10** to be transferred to the host computer **40** in accordance with an assigned priority. When a user requests the host computer to operate on a particular untransferred image file, the Work Queue Manager **82** causes the image file transfer to be interrupted and returns the operation to transfer the remaining portion of the untransferred image files to the host computer **40** after the user requested image file is transferred.

[0039] The Work Queue Manager **82** also performs the following functions:

[0040] Passes requests between the Camera Layer **84** and the layers directly above the Work Queue Manager **82**;

[0041] Serializes work requests;

[0042] Provides buffers for the transfer of parameter and request data between other layers;

[0043] Routes requests to the proper software component; and

[0044] Manages callback information/asynchronous requests.

[0045] Once the Camera Layer **84** notifies the Camera Manager **80** of a new connection of a digital camera **10** to the host computer **40**, the Camera Manager **80** can query for camera traits as follows:

[0046] Whether the digital camera **10** is a read only device;

[0047] Whether the digital camera **10** is able to delete files;

[0048] Whether the digital camera **10** is able to rename files (without copying and deleting the file);

[0049] If the digital camera **10** is only capable of reading whole files;

[0050] If the digital camera **10** supports taking a picture while connected; and

[0051] If the digital camera **10** contains support for the CoolFS module.

[0052] A Protocol Layer **86** is a software component which interprets generic commands which are passed through the Camera Layer **84** into commands that are understood by the camera firmware located in the Flash EPROM **29** (shown in **FIG. 1**). The Protocol Layer **86** determines what the true functionality of the digital camera **10** is (i.e., either interprets the generic command into the camera specific command(s) or returns an error). Once the Protocol Layer **86** interprets the requests into the camera specific commands, it then calls generic interface methods such as TransportSend and TransportReceive on a Transport Layer **88** software component. This makes it possible to provide support for a newly connected digital camera **10** by creating only a protocol layer for that newly connected digital camera **10** without making any changes to the other layers (provided the required transport layer is already in place from a previously connected camera). The Protocol

Layer **86** is also the layer that exposes an image file with embedded sound as two separate files (one file being the image data, and the other file being the audio segment data).

[**0053**] A Transport Layer **88** is the software component responsible for packaging the protocol specific commands into the required transport format (e.g., Win95/Win98 Serial, WINNT Serial, or USB) so they can be transmitted via the interface cable **36** to the host interface **26** of the digital camera **10**.

[**0054**] The following is a review of the operation of accessing digital camera files from a host computer in accordance with the present invention:

[**0055**] The user connects the digital camera **10** to the host computer **40** using the host interface **26**. The system detects that a camera device gets plugged in and notifies the Camera Layer **84** of the new device. Each time a digital camera **10** is connected to the host computer **40**, a camera alias is created in the Protocol Layer **86**. The camera alias acts as an address of the digital camera **10**.

[**0056**] Information about the interface **44** is cached away by the Camera Layer **84**, and the Camera Layer **84** creates the unique camera alias for use in communication with the Camera Manager **80**. This unique camera alias allows the Camera Manager **80** to specify what digital camera **10** it wishes to communicate with.

[**0057**] The Camera Layer **84** does a lookup in the registry to determine what protocol driver in the Protocol Layer **86** corresponds to the Vendor ID/Product ID for the digital camera **10** which was attached and then loads that driver. Since the transport already knows that the digital camera **10** was detected, the following layers shown in **FIG. 2** are now brought into operation: the Camera Layer **84**, the Protocol Layer **86**, and the Transport Layer **88**.

[**0058**] The Camera Layer **84** sends a "Camera Arrival" signal, including the unique camera alias, to the Camera Manager **80** indicating that the digital camera **10** has been detected. The Camera Manager **80** then creates it's own software model of the camera to maintain information such as the camera's unique alias (as known by the Camera Layer **84**), as well as the drive identification alphabetic letter (as known by the IFS **70**). This is one example of how the Camera Manager **80** connects the software components together.

[**0059**] The Camera Manager **80** then informs the IFS **70** that a new file system device has been installed. The IFS **70** obtains a new drive identification alphabetic letter from the system, and then asks the Camera Layer **84** for the contents of the drive.

[**0060**] Directory information is retrieved by the Camera Layer **84** and sent back to the IFS **70** (as AddFile and AddFolder messages). The choice of whether to recursively send file and folder information or wait for the IFS **70** to request it depends upon the various priorities embodied in the Camera Layer **84**.

[**0061**] A Client Application **73** sends a command through the User Mode API **72**, or through the Win32 API **71**. If the request came through the Win32 API **71**, then it is transmitted through the IFS **70** software component.

[**0062**] Camera modeling information is retrieved from the Camera Manager **80** and the request is passed to the unique camera alias on the Work Queue Manager **82**. The request is picked up by the Camera Layer **84** which uses its own camera modeling information to determine what protocol driver in the Protocol Layer **86** to forward the request to.

[**0063**] The protocol driver then reformats the generic Camera Layer **84** command into camera specific command(s). Such commands can include determining what images are stored in the removable memory card **30** of the digital camera **10** and reading information from the removable memory card **30**, setting the date and time properties of the digital camera **10**. The camera specific command(s) are then sent to the Transport Layer **88** which sends them via the interface cable **36** to the digital camera **10**. Any reply from the camera is retrieved and sent back up the chain in reverse order.

[**0064**] The following is used to transfer image files from the removable memory card **30** of the digital camera **10** to the host computer **40** in accordance with an assigned priority and permit the interruption of such transfer to operate on an untransferred image. When the digital camera **10** is connected to the host computer **40** and there are no user commands or OS file requests, low priority read requests are assigned by the Camera Manager **80** and cause the low priority image files to be transferred during idle time on the host computer **40**. This process is also known as prefetching. The host computer **40** transfers the image files one at a time from the removable memory card **30** of the digital camera **10** to the host computer memory (i.e., RAM **58** or the hard drive **56**).

[**0065**] When the Client Application **73** software component requests that an image file in the digital camera **10** be transferred, there are two conditions. First, if the requested image file has been prefetched or already transferred to the host computer **40**, the prefetched image file can be rapidly accessed and displayed from either the hard drive **56** or the RAM memory **58**. It is not necessary to transfer the image file from the digital camera **10** to the host computer **40** over the limited bandwidth interface which would provide a slower response time to the user. In this instance, the IFS **70** software component appears to be making a request to retrieve data from a file stored in the digital camera **10**, but it is actually retrieving the file from the hard drive **56** or RAM memory **58** of the host computer **40**.

[**0066**] In the second condition, the digital camera **10** is in the process of transferring low priority image files to the host computer **40** and the particular image file requested has not yet been transferred. In this instance, the IFS **70** software component sends a higher priority request to the Work Queue Manager **82**, and the Camera Layer **84** responds to the Work Queue Manager **82** and recognizes a higher priority request. The Work Queue Manager **82** causes the interruption of the low priority image file transfer and requests the Camera Layer **84** to operate on the particular higher priority untransferred image file. After this operation is completed in a manner discussed above, the Work Queue Manager **82** returns to its lower priority requests and continues the transfer of the low priority image files.

[**0067**] If the user disconnects the digital camera **10** from the host computer **40**, the Camera Layer **84** is notified about this disconnection and blocks all requests targeted for that digital camera **10**. The Camera Layer **84** then sends a "Camera Departure" signal to the Camera Manager **80**. Note

that there generally are two common serial transports associated with digital cameras. They are referred to as RS-232 and Universal Serial Bus (USB). In the case of an RS-232 transport, the Camera Layer **84** detects the disconnection by a failure of communication. The Camera Manager **80** then notifies the IFS **70** of the disconnect, and the IFS **70** removes the drive identification alphabetic letter from the host computer **40**. The Camera Layer **84** then completes the remaining requests for the digital camera **10** from the Work Queue Manager **82**.

[**0068**] **FIG. 3A** illustrates an exemplary view of a screen on a display monitor **52** of the host computer **40** after the digital camera **10** is connected to the host computer **40**. This illustrates that the digital camera **10** appears as another drive in the host computer **40**. For example, as shown in **FIG. 3A**, A: is a floppy drive, C: and D: are hard drives, E: is a CD-ROM, and F: is the DC290 digital camera. The left side of the screen shows the storage capacity of the digital camera **10** and how much storage is currently consumed, which is what the user would see for any drive on the host computer **40**.

[**0069**] **FIG. 3B** illustrates an additional exemplary view of a screen on the display monitor **52** of the host computer **40** depicting icons which represent files (including image data and audio segment data) and directories. The shown representative directory is an album.

[**0070**] In accordance with the present invention, the image files stored in the digital camera **10** can be accessed (e.g., viewed, copied or deleted) over a network, including the internet, and the digital camera **10** can be controlled via the network (e.g., take a new image by "selecting" an appropriate "current picture" file icon). The file current.jpg depicted in **FIG. 3B** enables a user to take a new picture from either the host computer **40** or one of the network computers (i.e., computer **62** or computer **64** shown in **FIG. 1**) when the digital camera **10** is connected via the host interface **26** to the host computer **40**. When the user opens the file current.jpg, the Win32 API **71** provides this user request to the IFS **70** software component. This request is passed down through the various software layers to the digital camera **10**.

[**0071**] When the user opens the current.jpg file on the host computer **40**, the digital camera **10** exposes a new image onto the image sensor **14**, processes the image, and temporarily stores the processed image in the RAM memory **28**. The Protocol Layer **86** monitors the digital camera **10**, which responds when the image has been captured and stored. The image is then automatically transferred from the digital camera **10** to the host computer **40** and stored in the host computer's memory (e.g., RAM **58** or hard drive **56**), and displayed to the user on the display monitor **52** of the host computer **40**. Thus, by simply opening the particular image file, the user is able to instruct the digital camera **10** to capture a new image and to immediately display the captured image on the host computer **40** either locally or via the network.

[**0072**] In accordance with the present invention, the host computer **40** identifies and selectively transfers at least one image file captured by the digital camera **10** which includes a digital image and at least one audio data segment. After storing the captured image file in the removable memory card **30** of the digital camera **10**, the digital camera **10** is connected to the host computer **40** via the interface cable **36**. The host computer **40** then identifies the captured image file and recognizes the presence of the digital image and the audio data segment stored in the removable memory card **30**.

[**0073**] As shown in **FIG. 3B**, at least two icons are provided on the display monitor **52** of the host computer **40** which respectively represent the digital image and the audio data segment. The digital image file is shown as file P0000047.jpg. Even though P0000047.jpg is a single file, the audio data segment is represented as a separate wave file P0000047.wav along with the image file. A user can then selectively access the digital image icon or the audio data segment icon to cause the digital image or the audio data segment to be transferred from the removable memory card **30** of the digital camera **10** to the host computer **40**.

[**0074**] **FIG. 4A** illustrates the image file structure when no audio has been recorded with the image (for example, file P0000046.jpg in **FIGS. 3B**). **FIG. 4B** illustrates the image file structure when audio has been recorded with the image. This is shown as file P0000047.jpg in **FIG. 3B**. Even though P0000047.jpg is a single file, the audio gets exposed as a separate wave file P0000047.wav along with the image file as shown in **FIG. 3B**.

[**0075**] In accordance with the present invention, when the user opens an audio data segment of a particular image file, i.e., if the user double clicks on the P0000047.wav icon of **FIG. 3B** in order to use their default sound player to play back this audio file, the Win32 API **71** issues a read request to the IFS **70**. The two transfer conditions previously specified are then invoked to determine how to operate on the requested image file. If the audio data segment file has not yet been transferred to the host computer **40**, then the Protocol Layer **86** receives the read request and determines the associated image file. The Protocol Layer **86** then reads only the audio data segment from the corresponding image file on the removable memory card **30** of the digital camera **10** and reformats it to be in the appropriate form.

[**0076**] The invention has been described in detail with particular reference to certain preferred embodiments thereof, but it will be understood that variations and modifications can be effected within the spirit and scope of the invention.

Parts List

[**0077**] **10** digital camera

[**0078**] **12** lens

[**0079**] **14** image sensor

[**0080**] **16** A/D converter

[**0081**] **18** processor

[**0082**] **19** microphone

[**0083**] **20** memory card interface

[**0084**] **21** audio amplifier and A/D converter

[**0085**] **22** image display

[**0086**] **24** user buttons

[**0087**] **26** host interface

[**0088**] **28** RAM memory

[0089] **30** removable memory card

[0090] **36** interface cable

[0091] **40** host computer

[0092] **42** memory card reader

[0093] **44** camera interface

[0094] **46** CD-ROM drive

[0095] **48** compact disc

[0096] **50** central processing unit

[0097] **52** display monitor

[0098] **54** keyboard

[0099] **55** mouse

[0100] **56** hard drive

[0101] **58** random access memory

[0102] **60** network card

[0103] **62** second computer

[0104] **64** third computer

Parts List Cont'd

[0105] **66** hardcopy printer

[0106] **70** Installable File System

[0107] **71** Win32 application programmers interface

[0108] **72** user mode application programming interface

[0109] **73** Client Application

[0110] **80** Camera Manager

[0111] **82** Work Queue Manager

[0112] **84** Camera Layer

[0113] **86** Protocol Layer

[0114] **88** Transport Layer

[0115] **100***a* image file

[0116] **100***b* image file

What is claimed is:

1. A method for a host computer to selectively operate on a plurality of image files captured by a digital camera, each image file having at least one digital image, the method comprising the steps of:

(a) storing the plurality of captured image files in a memory in the digital camera;

(b) coupling the digital camera memory to the host computer so that the host computer identifies the plurality of captured image files;

(c) the host computer identifying the digital camera memory as though it were a file system of an additional hard drive memory for accessing the captured image files; and

(d) accessing and selectively transferring the captured image files from the digital camera memory to the host computer.

2. The method of claim 1 wherein the coupling of the digital camera memory to the host computer is provided over a network.

3. The method of claim 2 wherein the network is the Internet.

4. The method of claim 2 wherein the host computer actuates the digital camera via the network to control the operation of the digital camera to capture an image.

5. A method for a host computer to selectively operate on at least one image file captured by a digital camera, the at least one image file having at least one digital image, the method comprising the steps of:

(a) storing the at least one captured image file in a memory in the digital camera;

(b) coupling the digital camera memory to the host computer so that the host computer identifies the at least one captured image file;

(c) the host computer identifying the digital camera memory as though it were a file system of an additional hard drive memory for accessing the at least one captured image file; and

(d) accessing and selectively transferring the captured image file from the digital camera memory to the host computer if the captured image file has not previously been transferred, but if the captured image file was previously transferred, accessing the transferred image file from the host computer.

6. A method for a host computer to selectively operate on at least a portion of an image file captured by a digital camera, the image file having at least one digital image, the method comprising the steps of:

(a) storing the captured image file in a memory in the digital camera;

(b) coupling the digital camera memory to the host computer so that the host computer identifies the captured image file;

(c) the host computer identifying the digital camera memory as though it were a file system of an additional hard drive memory for accessing the captured image file; and

(d) accessing a desired portion of the captured image file and selectively transferring such desired portion from the digital camera memory to the host computer if such desired portion has not previously been transferred, but if the desired portion was previously transferred, accessing such desired portion of the captured image from the host computer.

* * * * *