(12) UK Patent (19) GB (11) 2464049 (13) B

(45) Date of B Publication 05.12.2012

(54) Title of the Invention: System for identifying content of digital data

(51) INT CL: ***G06F 21/00*** (2006.01) ***G06F 17/30*** (2006.01)

(21) Application No: **1001473.6**

(22) Date of Filing: **28.07.2008**

Date Lodged: **29.01.2010**

(30) Priority Data:
(31) **11829662** (32) **27.07.2007** (33) **US**

(86) International Application Data:
PCT/US2008/009127 En 28.07.2008

(87) International Publication Data:
WO2009/017710 En 05.02.2009

(43) Date of Reproduction by UK Office **07.04.2010**

(56) Documents Cited:
EP 0349106 A1
US 6675174 B1
US 20070074147 A1
US 20030061490 A1
US 6968337 B2
US 4677466 A
US 20050216433 A1

(58) Field of Search:
As for published application 2464049 A viz:
INT CL **G06F**
Other: **USPC: 707/1, 6, 7, 104.1, E17.009, E17.011;**
Online: PubWest, Internet;
updated as appropriate

Additional Fields
INT CL **G06K**
Other: **Online: WPI, EPODOC**

(72) Inventor(s):
Erling Wold

(73) Proprietor(s):
Audible Magic Corporation
985 University Avenue #35, Los Gatos 95032, California, United States of America

(74) Agent and/or Address for Service:
WP Thompson
Coopers Building, Church Street, Liverpool, L1 3AB, United Kingdom

GB 2464049 B

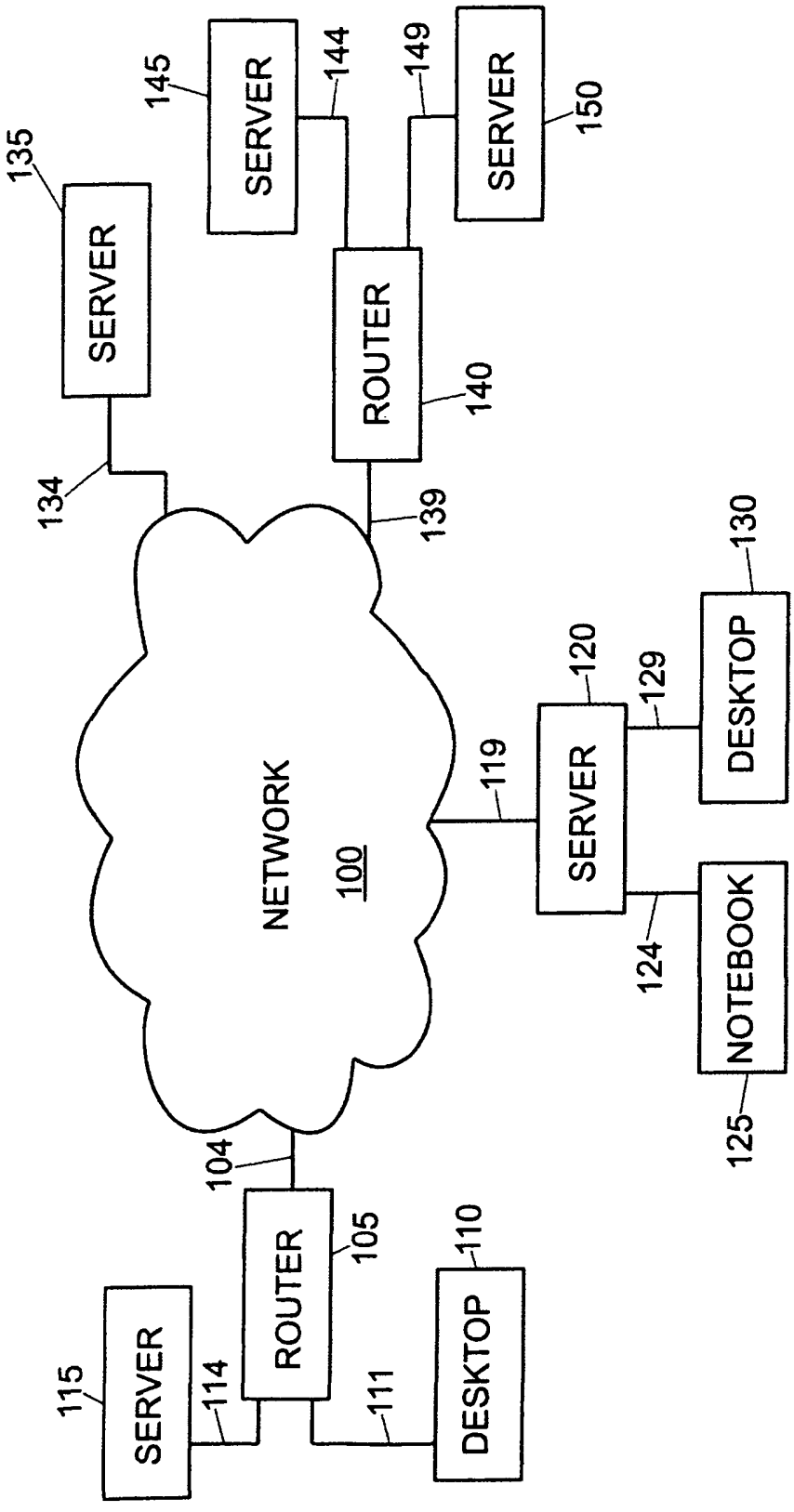

SERVER
135
SERVER
145
144
SERVER
149
150
ROUTER
140
134
139
NETWORK
100
104
119
SERVER
120
129
DESKTOP
130
124
NOTEBOOK
125
SERVER
115
114
ROUTER
105
111
DESKTOP
110

FIG. 1

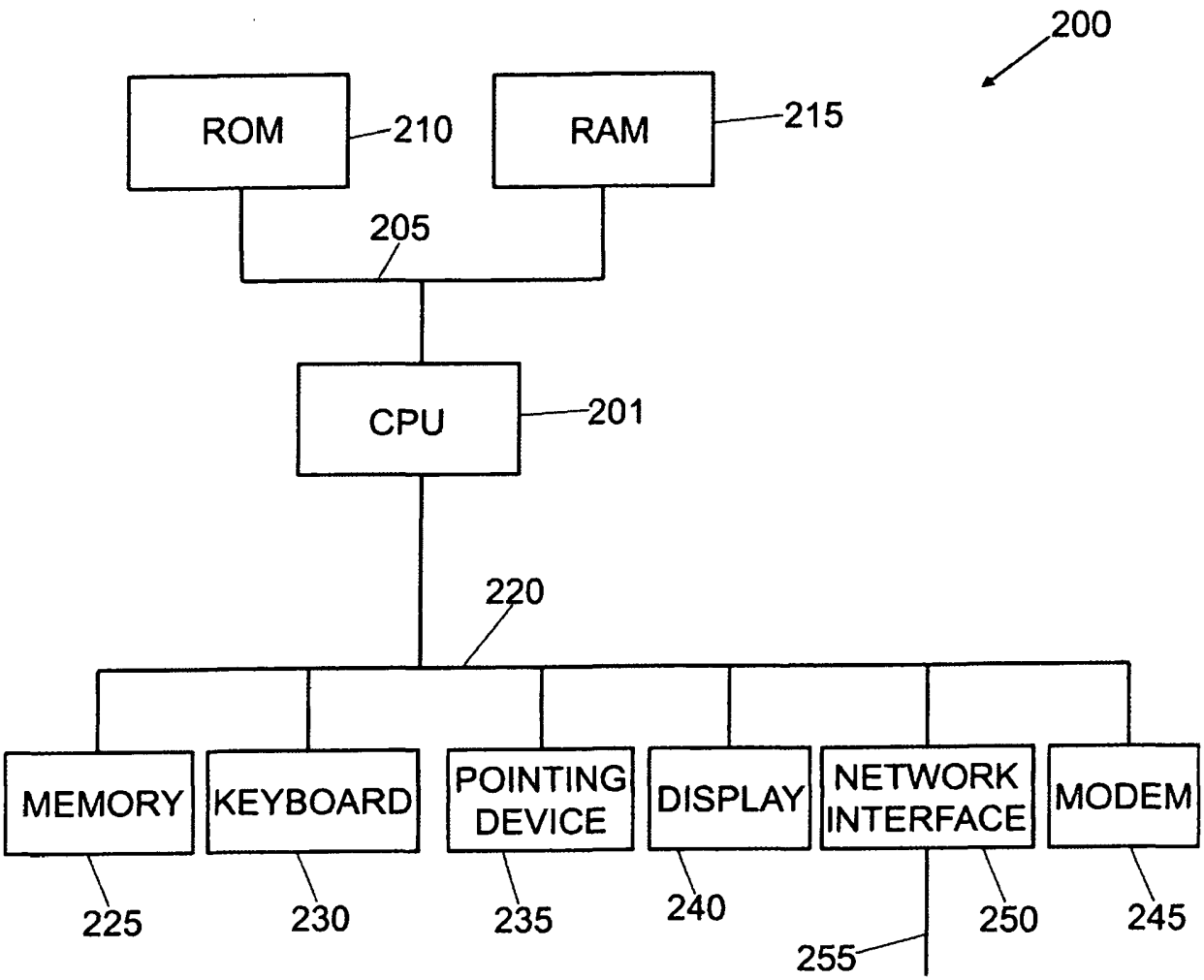

200
ROM
210
RAM
215
205
CPU
201
220
MEMORY
KEYBOARD
POINTING DEVICE
DISPLAY
NETWORK INTERFACE
MODEM
225
230
235
240
255
250
245

FIG. 2

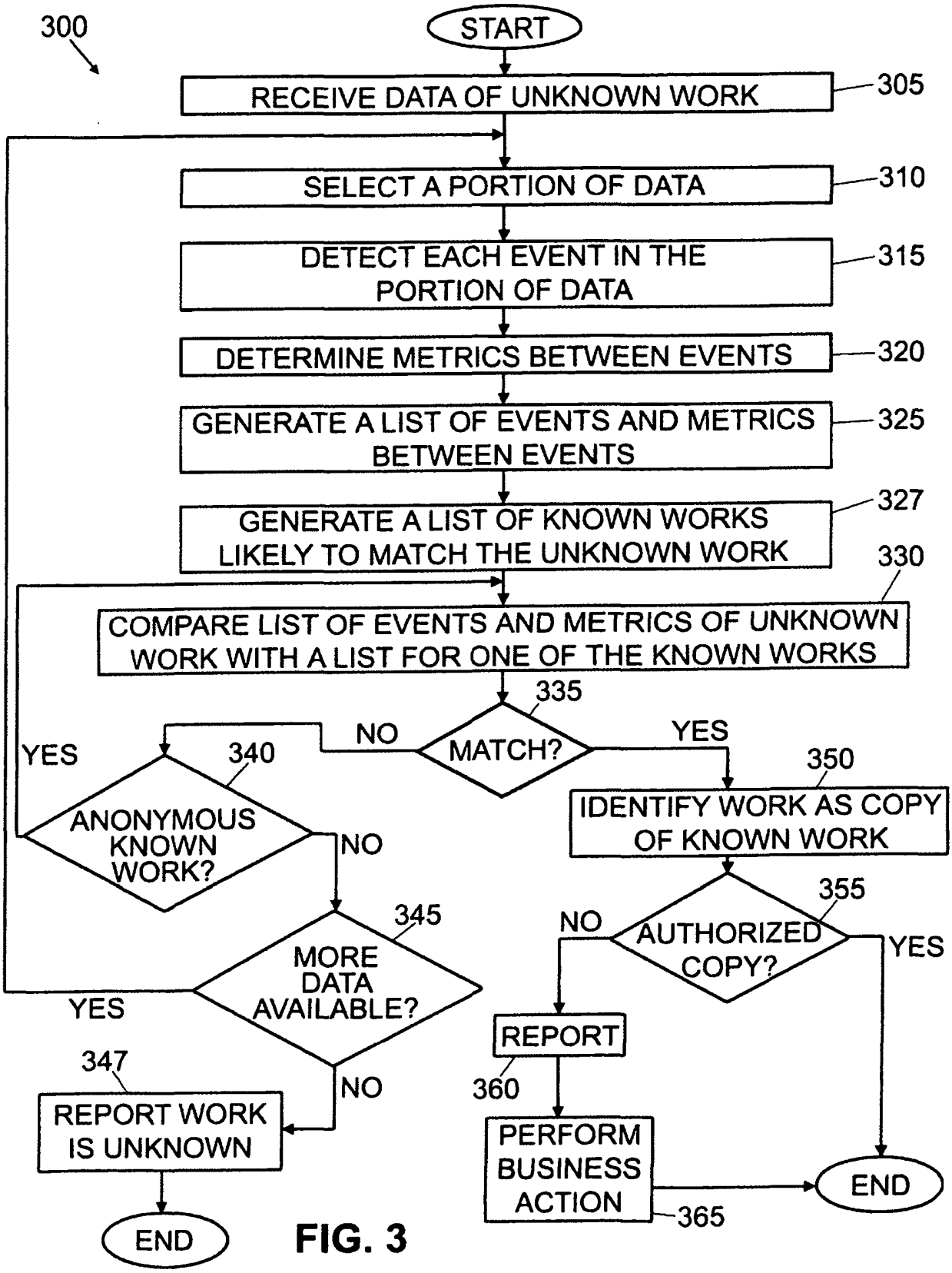

300
START
RECEIVE DATA OF UNKNOWN WORK
305
SELECT A PORTION OF DATA
310
DETECT EACH EVENT IN THE PORTION OF DATA
315
DETERMINE METRICS BETWEEN EVENTS
320
GENERATE A LIST OF EVENTS AND METRICS BETWEEN EVENTS
325
GENERATE A LIST OF KNOWN WORKS LIKELY TO MATCH THE UNKNOWN WORK
327
330
COMPARE LIST OF EVENTS AND METRICS OF UNKNOWN WORK WITH A LIST FOR ONE OF THE KNOWN WORKS
335
MATCH?
NO
YES
340
ANONYMOUS KNOWN WORK?
YES
NO
345
MORE DATA AVAILABLE?
YES
NO
347
REPORT WORK IS UNKNOWN
END
350
IDENTIFY WORK AS COPY OF KNOWN WORK
355
AUTHORIZED COPY?
NO
YES
REPORT
360
PERFORM BUSINESS ACTION
365
END

FIG. 3

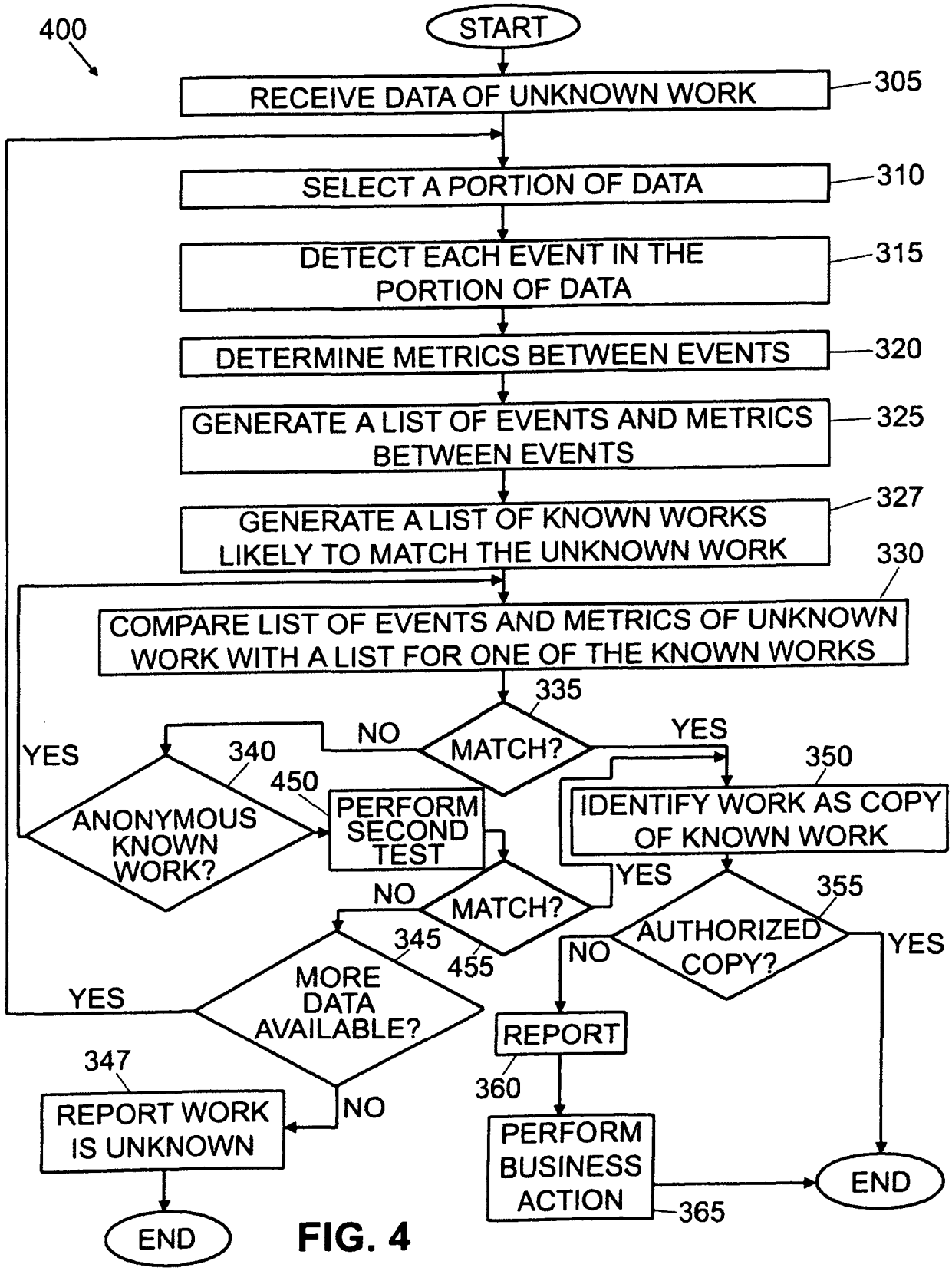

400
START
RECEIVE DATA OF UNKNOWN WORK
305
SELECT A PORTION OF DATA
310
DETECT EACH EVENT IN THE PORTION OF DATA
315
DETERMINE METRICS BETWEEN EVENTS
320
GENERATE A LIST OF EVENTS AND METRICS BETWEEN EVENTS
325
GENERATE A LIST OF KNOWN WORKS LIKELY TO MATCH THE UNKNOWN WORK
327
330
COMPARE LIST OF EVENTS AND METRICS OF UNKNOWN WORK WITH A LIST FOR ONE OF THE KNOWN WORKS
335
MATCH?
NO
YES
340
ANONYMOUS KNOWN WORK?
YES
450
PERFORM SECOND TEST
350
IDENTIFY WORK AS COPY OF KNOWN WORK
MATCH?
NO
YES
355
AUTHORIZED COPY?
NO
YES
345
455
MORE DATA AVAILABLE?
YES
REPORT
360
347
REPORT WORK IS UNKNOWN
NO
PERFORM BUSINESS ACTION
365
END
END

FIG. 4

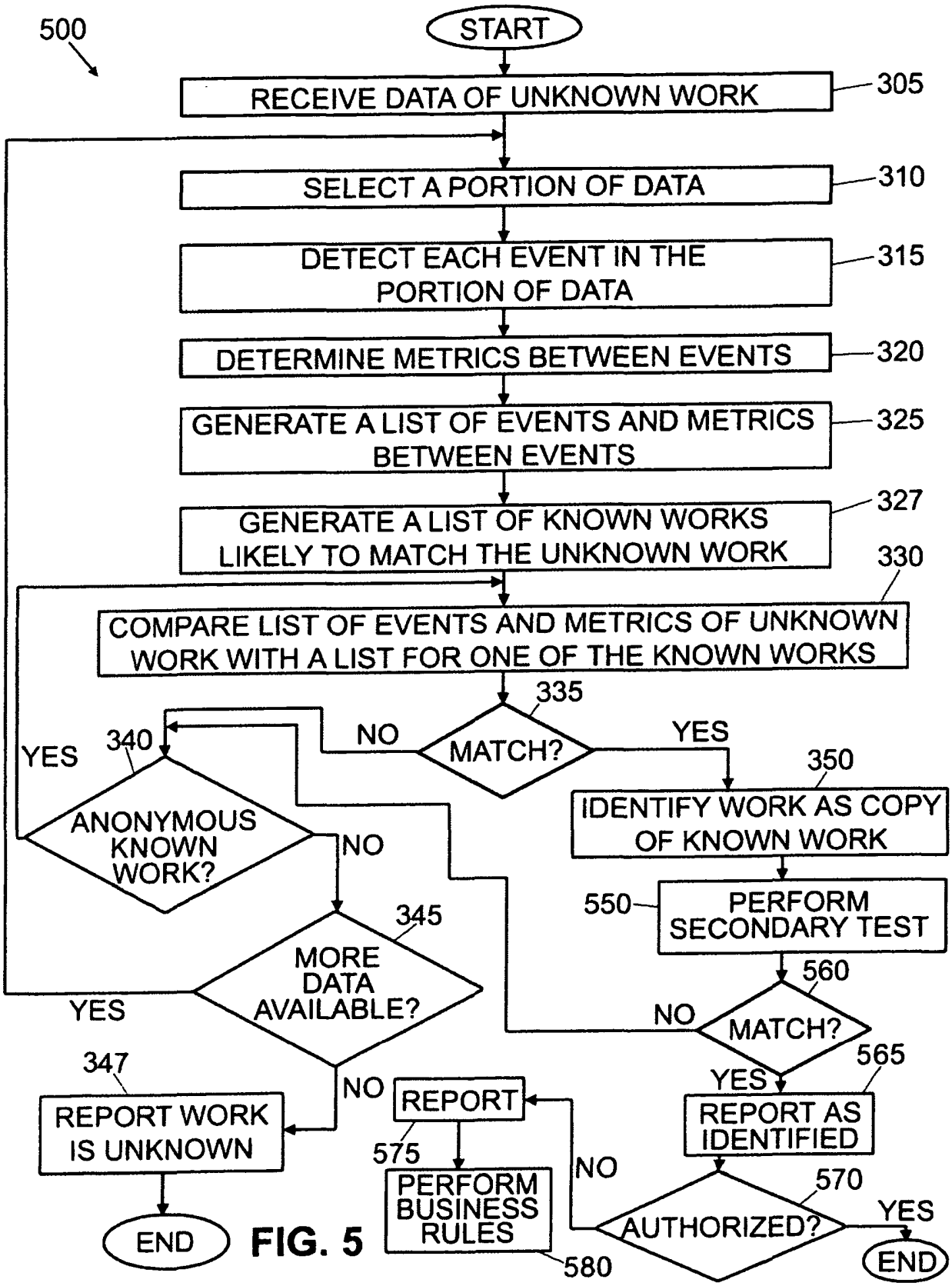

500
START
RECEIVE DATA OF UNKNOWN WORK
305
SELECT A PORTION OF DATA
310
DETECT EACH EVENT IN THE PORTION OF DATA
315
DETERMINE METRICS BETWEEN EVENTS
320
GENERATE A LIST OF EVENTS AND METRICS BETWEEN EVENTS
325
GENERATE A LIST OF KNOWN WORKS LIKELY TO MATCH THE UNKNOWN WORK
327
330
COMPARE LIST OF EVENTS AND METRICS OF UNKNOWN WORK WITH A LIST FOR ONE OF THE KNOWN WORKS
335
MATCH?
NO
YES
340
ANONYMOUS KNOWN WORK?
YES
NO
345
MORE DATA AVAILABLE?
YES
NO
347
REPORT WORK IS UNKNOWN
END
350
IDENTIFY WORK AS COPY OF KNOWN WORK
550
PERFORM SECONDARY TEST
560
MATCH?
NO
YES
565
REPORT AS IDENTIFIED
570
AUTHORIZED?
NO
YES
END
REPORT
575
PERFORM BUSINESS RULES
580

FIG. 5

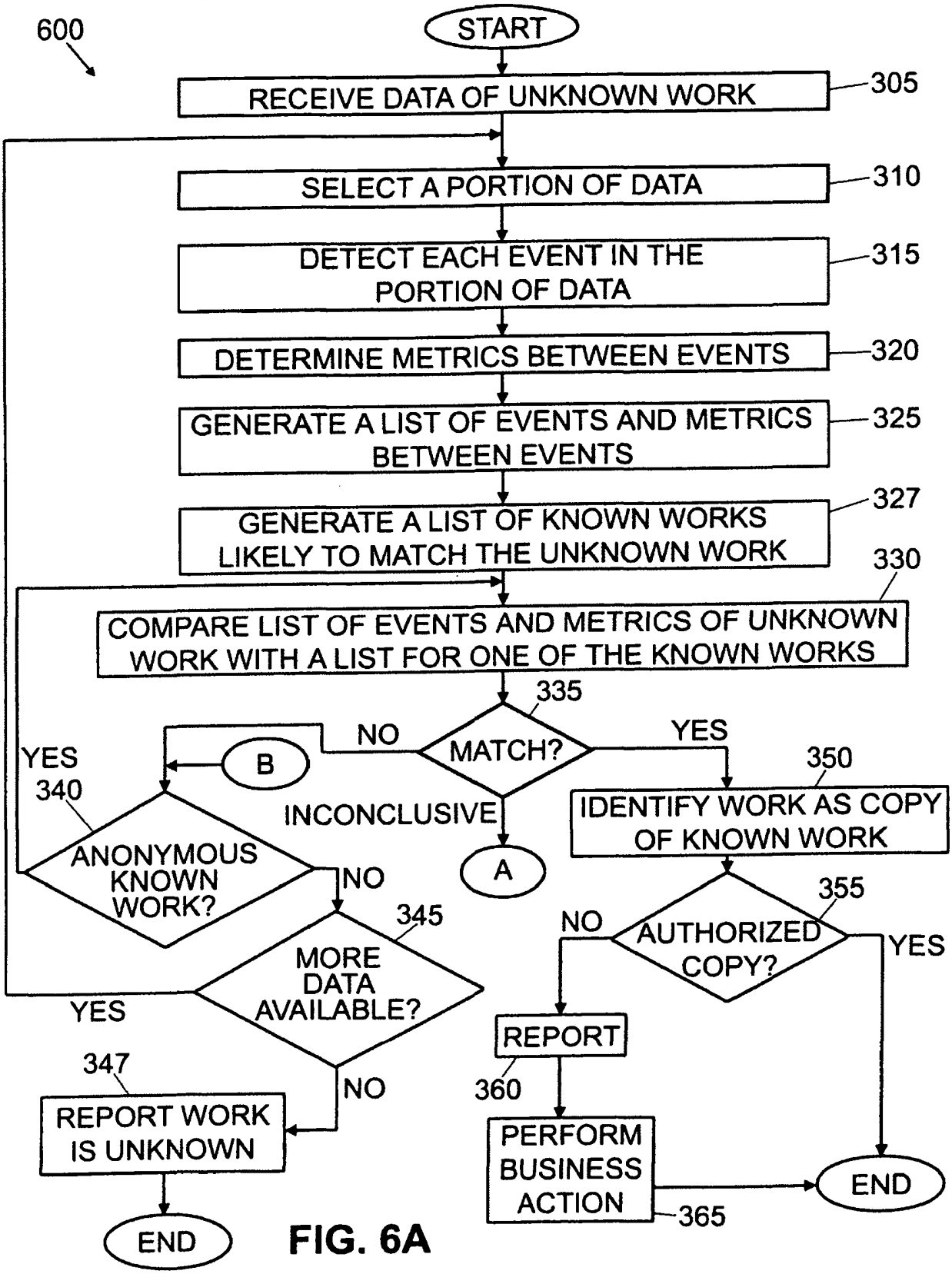

600
START
RECEIVE DATA OF UNKNOWN WORK
305
SELECT A PORTION OF DATA
310
DETECT EACH EVENT IN THE PORTION OF DATA
315
DETERMINE METRICS BETWEEN EVENTS
320
GENERATE A LIST OF EVENTS AND METRICS BETWEEN EVENTS
325
GENERATE A LIST OF KNOWN WORKS LIKELY TO MATCH THE UNKNOWN WORK
327
330
COMPARE LIST OF EVENTS AND METRICS OF UNKNOWN WORK WITH A LIST FOR ONE OF THE KNOWN WORKS
335
MATCH?
NO
YES
B
YES
340
ANONYMOUS KNOWN WORK?
NO
INCONCLUSIVE
A
350
IDENTIFY WORK AS COPY OF KNOWN WORK
355
AUTHORIZED COPY?
NO
YES
345
MORE DATA AVAILABLE?
YES
347
REPORT WORK IS UNKNOWN
NO
END
REPORT
360
PERFORM BUSINESS ACTION
365
END

FIG. 6A

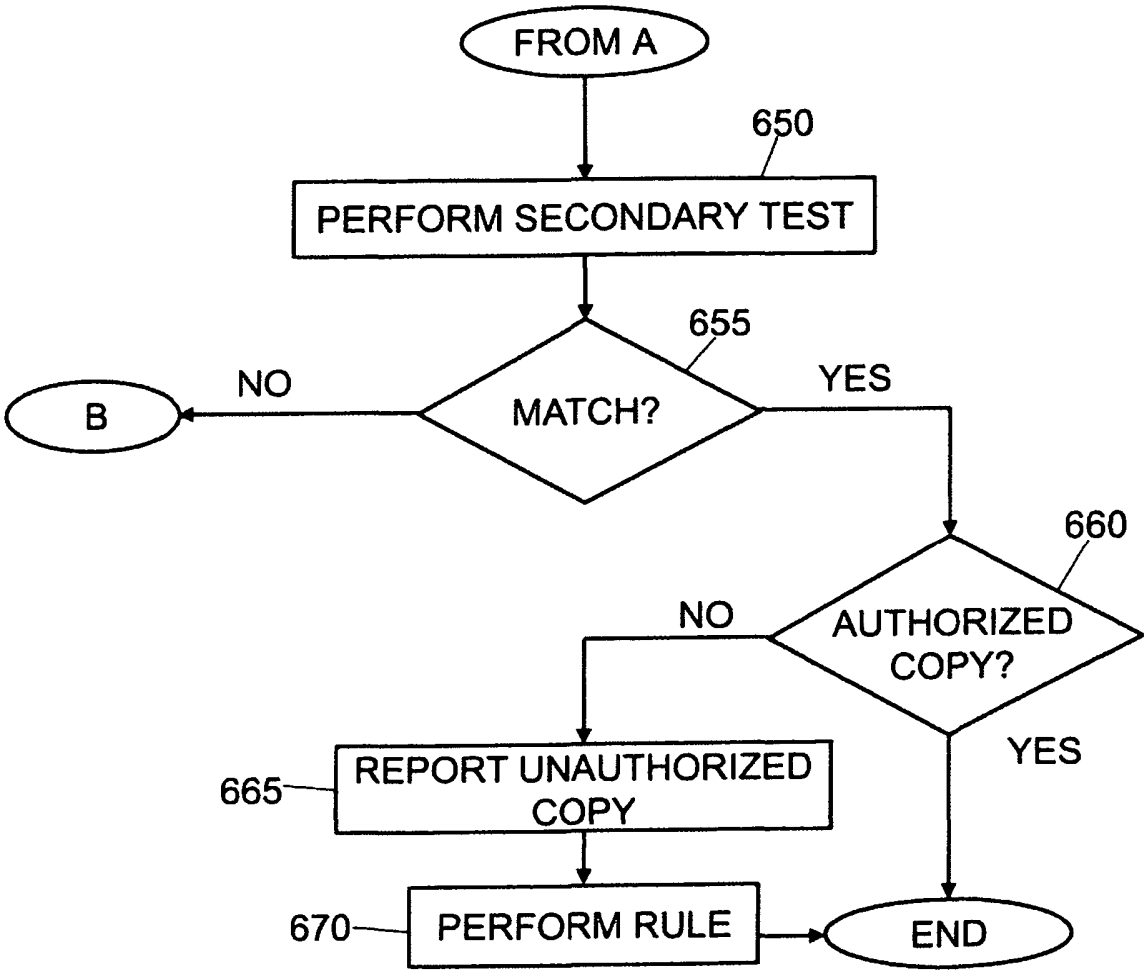

FROM A
650
PERFORM SECONDARY TEST
655
MATCH?
NO
B
YES
660
AUTHORIZED COPY?
NO
YES
665
REPORT UNAUTHORIZED COPY
670
PERFORM RULE
END

FIG. 6B

Figure 7
Outer loops of list matching process

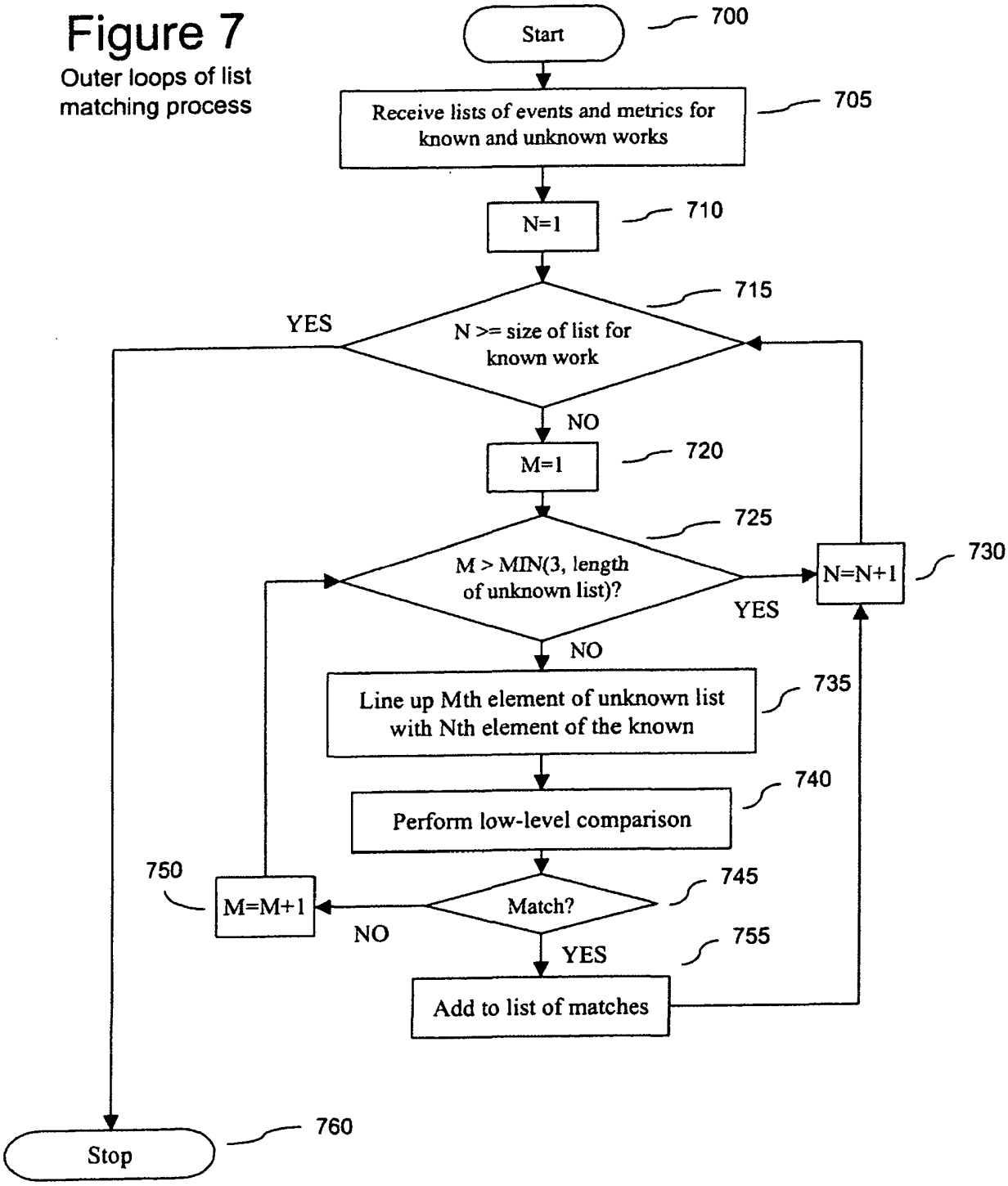


Figure 8
Low-level comparison

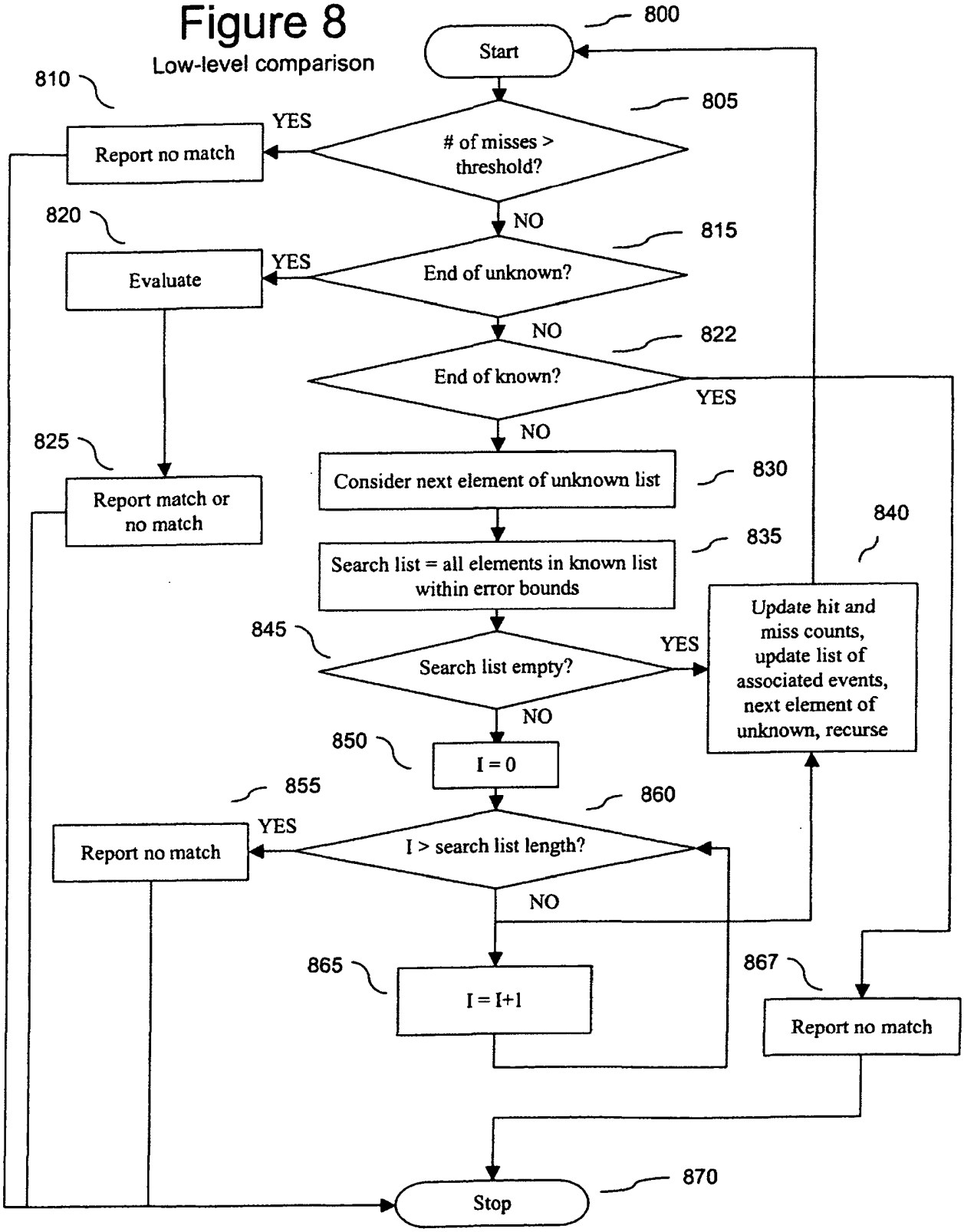
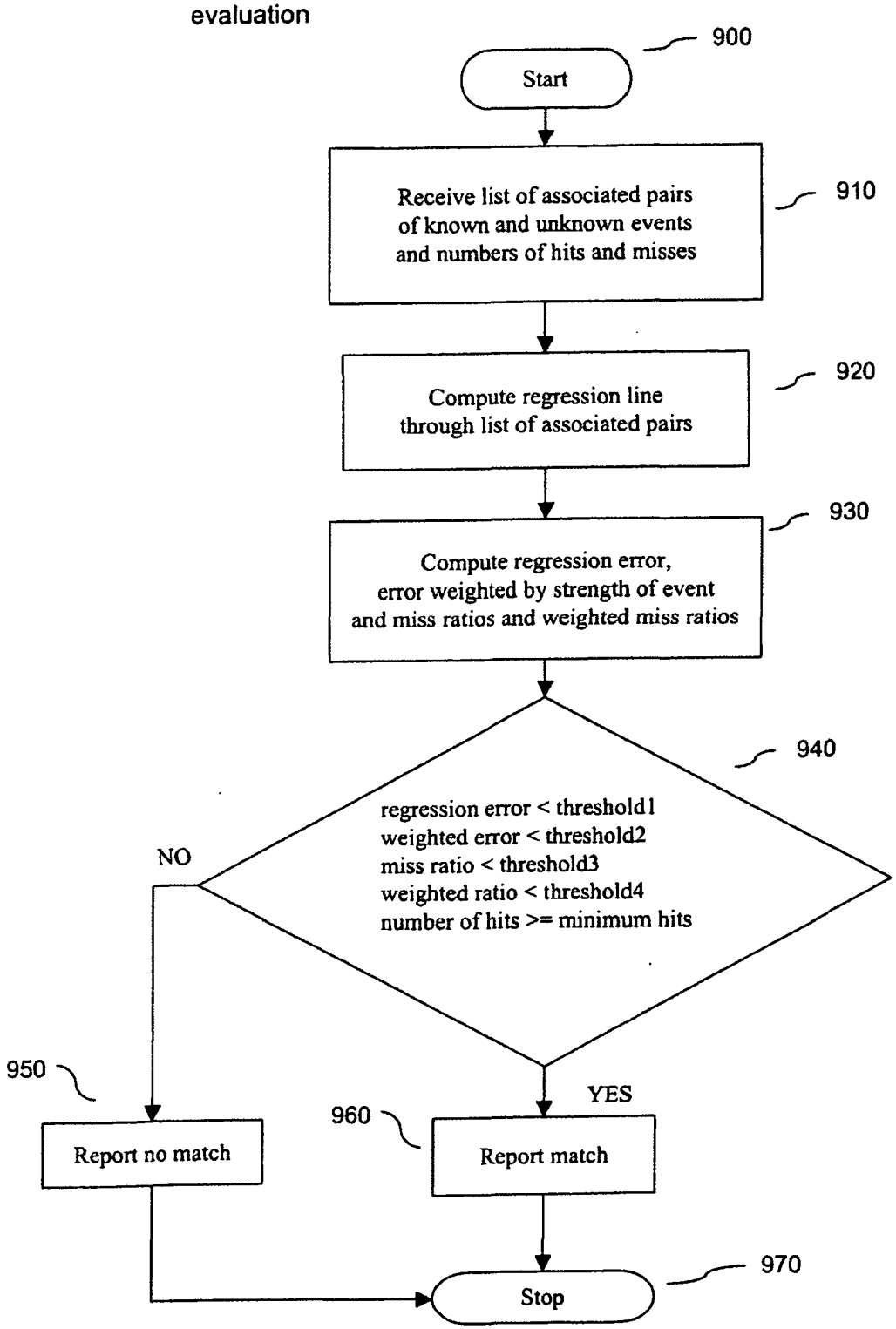


Figure 9
evaluation



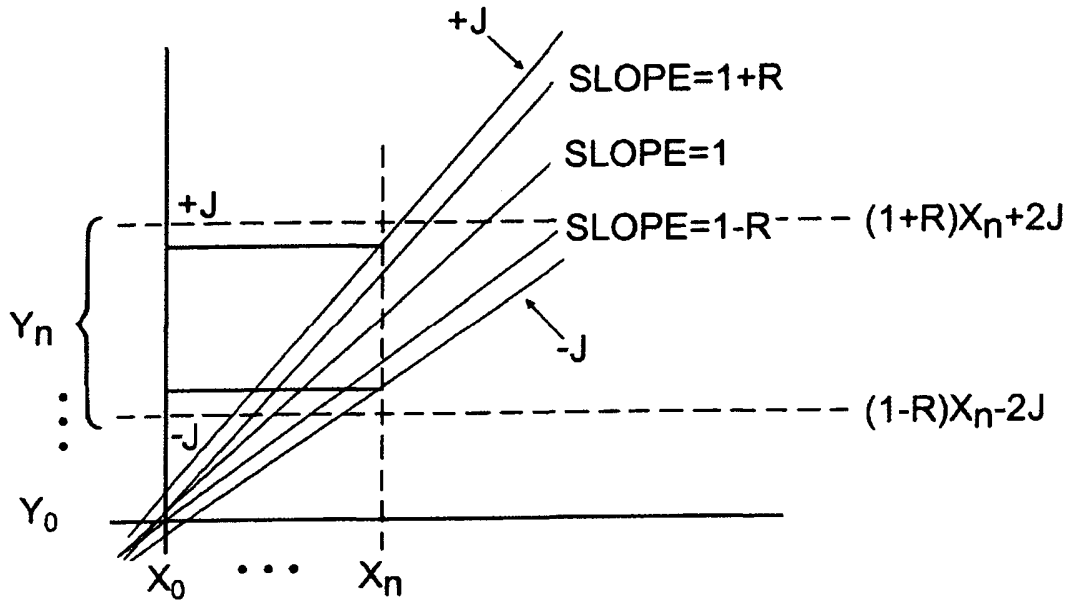


FIG. 10
ERROR BOUND #1

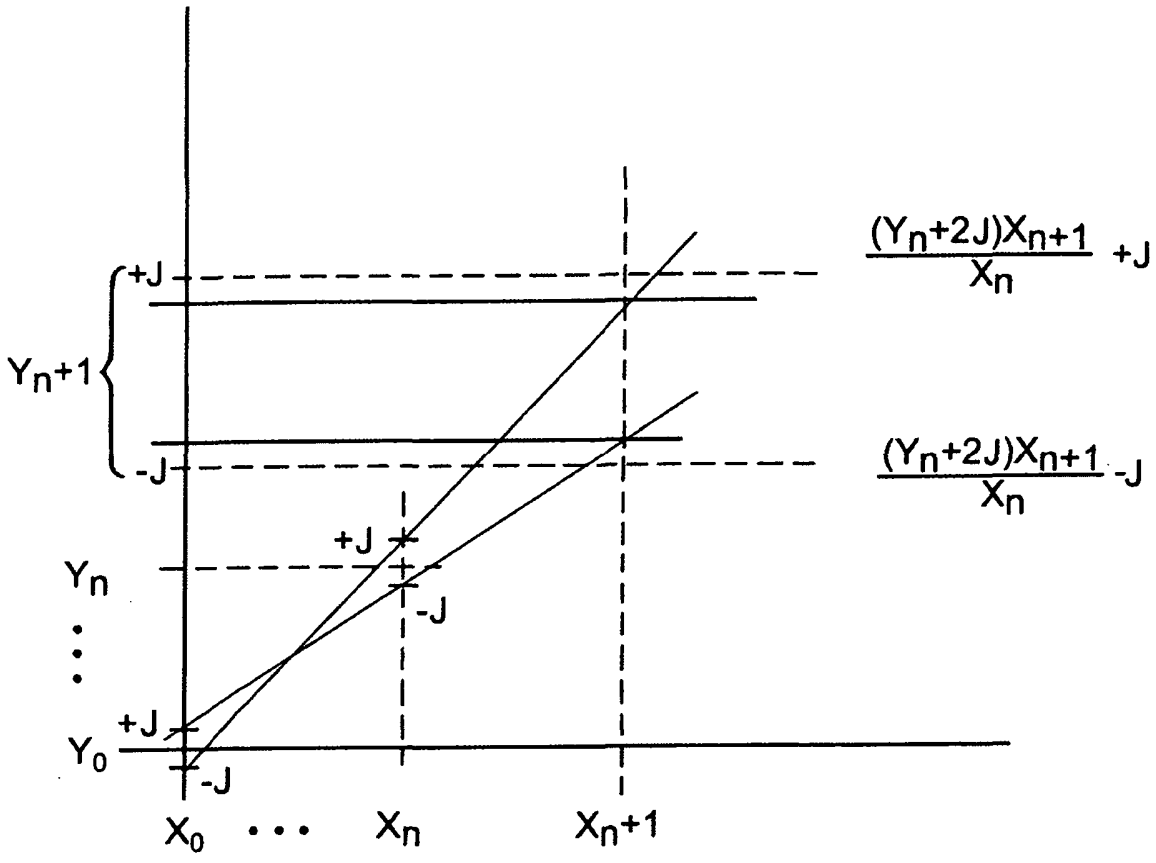
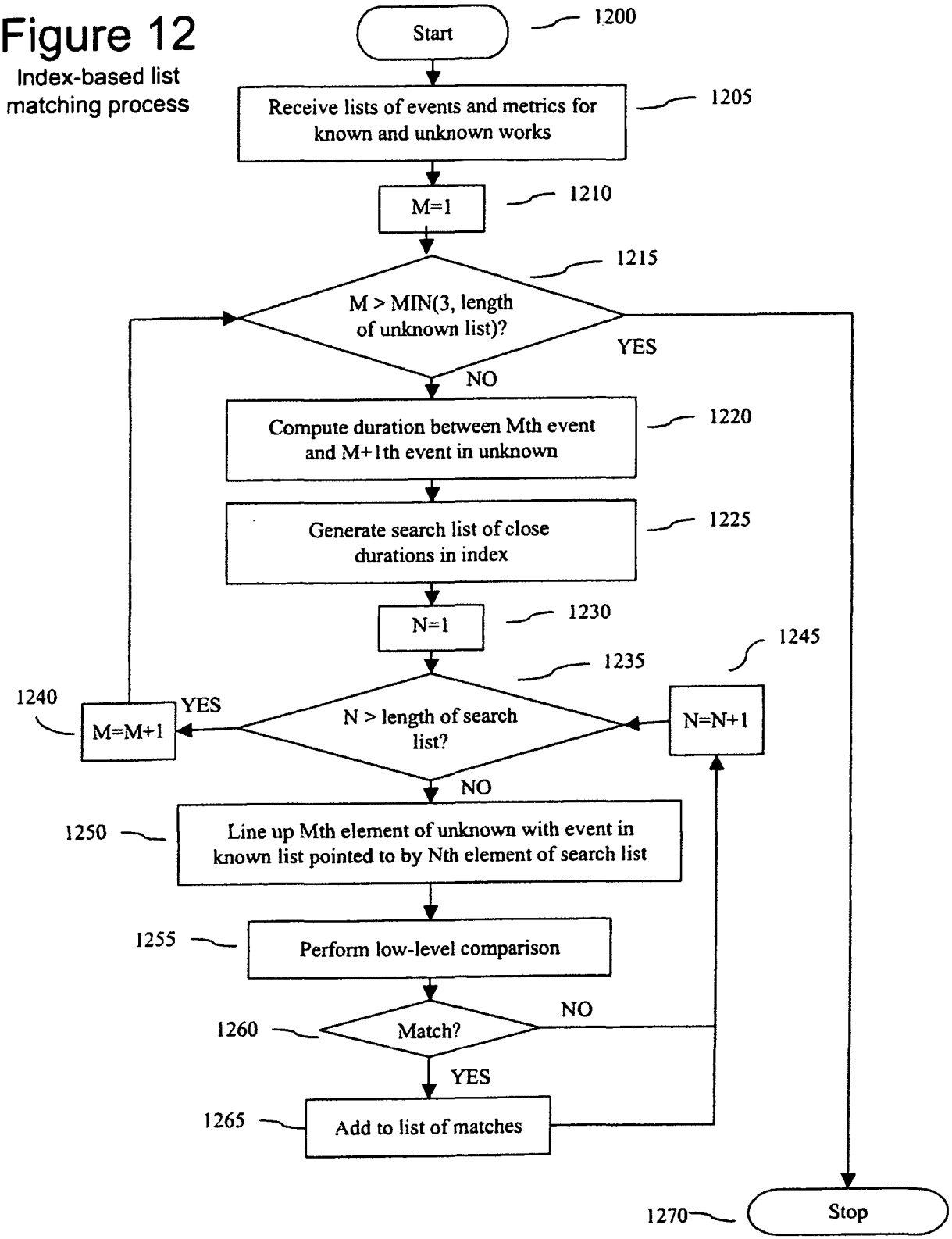


FIG. 11

ERROR BOUND #2

Figure 12
Index-based list matching process



SYSTEM FOR IDENTIFYING CONTENT OF DIGITAL DATA

FIELD OF THE INVENTION

[0001] This invention relates to identifying content of digital data. More particularly, this invention relates to identifying a work represented as digital data from an arbitrary segment of the digital data representing the content of the work in a memory of a digital processing unit. Still more particularly, this invention relates to detecting particular events in the segment of data and the time between events and comparing the time between events to the time between events in known works to determine the identity of the work that the segment of data represents.

PRIOR ART

[0002] In the past few years, the Internet has become a popular medium for distributing or in some other way transferring works between users. For purposes of this discussion, a work is any piece of art or work product that varies over time and is placed on a medium for distribution. More particularly, a work is a piece of art or work product that is placed in an electronic medium for use and/or distribution. Furthermore, a registered work is a work for which the identity of the work is known. Recently, many web sites have been offering more video works for viewing and transfers to viewers. For example, the web site YouTube.com (RTM) provides clips of video data that users submit for other viewers to download and view. For purposes of this invention, some of the clips submitted by viewers are a portion of a copyrighted work such as a television show or movie. Owners of copyrighted works are often not compensated by the website owners or the users for the reproduction of the work. Thus, owners of the works seek either to prevent these web sites for providing the clips of their works or receive compensation for reproduction of their works.

[0003] Also, as Digital Video Disks (DVDs) have become more popular, the downloading and unauthorized reproduction of video works has become a problem. There is a booming market for pirated or unauthorized reproductions of video works. In the past, makers of DVDs have tried to use encryption and other methods to prevent unauthorized reproduction of the works. However, most of the methods devised to prevent unauthorized use have been overcome or circumvented by hackers. Thus,

owners of the works seek ways to either detect the unauthorized work and receive compensation or prevent the reproduction.

[0004] In the past, those skilled in the art have made many advances in detecting the identity of audio works. One example, a reliable method for identifying audio works, is given in U.S. Patent Number 5,918,223 issued to Blum et al. (Blum) which is hereby incorporated by reference as if set forth herewith. In Blum, fingerprints of segments of audio data of an unknown work are generated and compared to fingerprints of data of known works until a match is found. The fingerprints can be one or more of any number of attributes of audio contents. Some examples of audio attributes include, but are not limited to pitch, frequency spectra, and mel-filtered cepstral coefficients. This method is very reliable in identifying audio works.

[0005] However, those skilled in the art have yet to find an efficient and reliable method for identifying video works. One method that has proven promising for identifying video works is the use of scene change events. Data for an unknown video work is read and the scene change events are detected. A metric such as the time between events or frames between the events is determined. For purposes of this discussion, a scene change event is one or more empty frames between different colored frames and/or significant changes in the visual between two adjacent frames or significant change in visual content over a small amount of time. Any event may be used as long as detection is easily repeatable. The sequence of metrics between the events of the unknown work are then compared to a list of metrics between events for known works until a sufficient match is made and the unknown work is identified.

[0006] There are several problems with the above-identified method of video work identification. The above method of identifying works is reliable when the unidentified work is a direct copy of an identified work. However, this method is not reliable when a copy of the video work is not a direct copy of the video work. For purposes of this discussion, a direct copy is a copy of a work that includes all of the visual data of the copied video work presented in the same manner as created by the owner of the work. Furthermore, an indirect copy is a copy of video work that has the data modified from the original work. Indirect copies can be made in many different manners. Some examples include but are not limited to reformatting of the data, such as from letter box to conventional format; recording the video work in a second medium,

such as video taping a movie from a scene at a theater; copying only a portion of the data; noise introduced in the routine broadcasting of the work; format conversions that occur such as telecining, digitizing, compressing, digital-analog-digital re-sampling, keystoning, rotation translation, playback rate changes, and the myriad of other common transformations commonly known in the art.

[0007] Typically, an indirect copy has a different video quality from the original work. Thus, some scene change events in an indirect copy may not be detected or other scene change events caused by the copying method may be detected. In addition, because of time scaling and/or playback rate changes, the time between detected events in the original work and the indirect copy may vary. Thus, the list of metrics for the unknown copy is different from the list for the original work and the above-described method may not detect the match.

[0008] Thus, those skilled in the art are constantly striving to find a new method that is more reliable for identifying the unidentified work.

SUMMARY OF THE INVENTION

[0009] In accordance with a first aspect of the present invention, there is a method for identifying a work in a digital processing system comprising:

selecting a portion of data of an unknown work;

detecting each event in said portion of data of said unknown work, wherein events are perceptual occurrences in a work that can be successively positioned in time;

determining an event metric between each successive event in said portion of data in said unknown work;

generating a first list of event metrics between said events for said unknown work;

receiving a second list of event metrics for a known work;

generating a third list comprising events in the first list that match events in the second list by

performing a comparison to determine whether an Mth event of the first list matches an Nth event of the second list, wherein M is an index over events in the first list and N is an index over events in the second list;

responsive to a determination of a match, adding the matching events to the third list and incrementing the index N;

responsive to a determination that there is not a match, incrementing the index M,

wherein the comparison is performed until the index N is greater than or equal to a number of events in the second list and the index M is greater than a threshold; and

determining said unknown work is a copy of said known work based on the third list comprising the events in the first list that match events in the second list.

[0009a] A first advantage of a system in accordance with this invention is that indirect copies can be identified with an improved confidence level. A second advantage of a system in accordance with this invention is that the system provides an efficient method to identify an unknown work that can be used during data transfers without unduly hampering the transfer of data between two digital processing systems. A third advantage of this invention is that even if the unidentified work is only a portion of an original work, it can be identified accurately. A fourth advantage of this invention is that this technique can be applied to match any series of events detected over time if the path to one detector introduces significant noise or errors into the stream of data.

[0010] Sometimes, the list for an unknown work must be compared to multiple lists of known works until a match is found. In some embodiments, the works most likely to match the unknown work are selected for testing. Sometimes more data than is needed to identify an unknown work is received. Thus, the system selects a portion of data from the segment to test. The received data may

come from data stored in a memory, data being broadcast over a network, or data being passed between systems connected over a network. Sometimes the data may also be read from a medium.

[0011] In some embodiments, the system determines whether the unknown work is an authorized copy of a known work when the unknown work is identified. The system may then generate a report stating whether the unknown work is an authorized or unauthorized copy of the known work. In some embodiments, a business rule is performed in response to a determination that the unknown work is an unauthorized copy. In some embodiments the business rule may direct that the copy be degraded in quality or altered in a plurality of ways.

[0012] In some embodiments, a secondary test is performed responsive to a determination that the unknown work is a copy of a known work from comparing lists. In other embodiments, a secondary test is performed on the unknown work when the results of the list comparisons are inconclusive. In still other embodiments, a secondary test is performed when no matches are found using the list comparisons.

[0013] In some embodiments of this invention, the comparison of the lists is performed in the following manner. First the system receives the list of metrics of the unknown work and the list of metrics of the known work. The system then determines a number of events in the list of metrics for the unknown work that match events in the list of metrics for the known work. The number of matches is then compared to a threshold. If the number of matches is greater than the threshold, a match between the lists is determined.

[0014] The comparison of elements of the list may be performed in the following manner. The system aligns an Mth element of the list of metrics of the unknown work with an Nth element of the list of metrics for the known work. The system then performs a first comparison of the lists starting from the Mth element in the list of metrics for the unknown list and the Nth element in the list of metrics

for the known work. The system then determines whether the lists match from the Mth element and the Nth element in the respective lists.

[0015] This may be repeated iteratively to determine whether there is a match. The lists starting from the Mth and Nth elements may be compared in the following manner. The lists may be compared one element at time starting from the Mth and Nth elements and the system determines the number of elements that match. This number is compared to a threshold and a match is determined if the number of matching elements surpasses the threshold. In some embodiments a number of misses is recorded and the lists are determined not to match if the number of misses exceeds a miss threshold.

[0016] In some embodiments the comparison of lists may begin by generating a list of associated pairs wherein each pair consists of an element from the list of metrics of the unknown work and a matching element from the list of metrics of the known work, where a matching element in the known work is an element whose metric is within an error tolerance of the metric for the associated element in the unknown work. A regression line is computed through the associated pairs in the new list. The regression error associated with this line is then used, along with the total number of hits and misses, to determine whether there is a match.

[0017] In other embodiments, a regression error is calculated from the computation of said regression line, the regression error is then compared to a regression error threshold. A match is determined when the regression error is less than the regression error threshold.

[0018] In other embodiments, a weighted error is calculated from the computation of the regression line. The weighted error is then compared to a weighted error threshold. A match is determined when the weighted error is less than the weighted error threshold.

[0019] In other embodiments, a miss ratio is calculated from the

computation of said regression line, the miss ratio is then compared to a miss ratio threshold. A match is determined when the miss ratio is less than the miss ratio threshold.

[0020] In other embodiments, a weighted miss ratio is calculated from the computation of the regression line, the weighted miss ratio is then compared to a weighted miss ratio threshold. A match is determined when the weighted miss ratio is less than the miss ratio threshold. In other embodiments, a plurality of the error measures described above may be used to determine a match.

[0020a] In accordance with a second aspect of the present invention, there is a system for identifying a digital work, the system being configured to:

select a portion of data of an unknown work;

detect each event in said portion of data of said unknown work, wherein events are perceptual occurrences in a work that can be successively positioned in time;

determine an event metric between each successive event in said portion of data in said unknown work;

generate a first list of event metrics between said events for said unknown work;

receive a second list of event metrics for a known work;

generate a third list comprising events in the first list that match events in the second list; and

determine said unknown work is a copy of said known work based on the third list comprising the events in the first list that match events in the second list, wherein the system is configured to generate the third list by

performing a comparison to determine whether an Mth event of the first list matches an Nth event of the second list, wherein M is an index over events in the first list and N is an indeed over events in the second list, adding the matching events to the third list and incrementing the index N responsive to a determination

of a match, and responsive to a determination that there is not a match, incrementing the index M, wherein the comparison is performed until the index N is no longer less than a number of events in the second list and the index M is greater than a threshold.

BRIEF DESCRIPTION OF THE DRAWINGS

[0021] The above and other features and advantages of a system in accordance with this invention are described in the Detailed Description below and are shown in the following drawings:

[0022] Figure 1 illustrating a block diagram of a network including at least one digital processor executing an identification system in accordance with this invention;

[0023] Figure 2 illustrating a block diagram of components of a digital processing system executing an identification system in accordance with this invention;

[0024] Figure 3 illustrating a block diagram of a first exemplary embodiment of an identification process in accordance with this invention;

[0025] Figure 4 illustrating a block diagram of a second exemplary embodiment of an identification process in accordance with this invention;

[0026] Figure 5 illustrating a block diagram of a third exemplary embodiment of an identification process in accordance with this invention;

[0027] Figures 6A and 6B illustrating a block diagram of a first exemplary embodiment of an identification process in accordance with this invention;

[0028] Figures 7-9 illustrating a block diagram of a first exemplary embodiment of a process for comparing a list of events and metrics from an unknown work with a list of events and metrics of known work in accordance with this invention; and

[0029] Figures 10 and 11 illustrating an exemplary embodiment of a method for constructing an error bound in the comparison of the events and metrics of an unknown work with the events and metrics of a known work in accordance with this invention; and

[0030] Figure 12 illustrating a block diagram of a second exemplary embodiment of a process for comparing a list of events and metrics from an unknown work with a list of events and metrics of known work in accordance with this invention.

DETAILED DESCRIPTION

[0031] This invention relates to identification of an unknown work from digital data representing the content of the unknown work. For purposes of this discussion, a work is any piece of art or work product that is placed on medium for distribution. More particularly, a work is a piece of art or work product that is placed in an electronic medium for use and/or distribution. Furthermore, a registered work is a work for which the identity of the work is known and the time between events in the data of the work is known.

[0032] This identification system is based upon measuring a metric between events in works and comparing the metric between events in a known work and an

unknown work. For purposes of this discussion, an event is a perceptual occurrence in a work that can be positioned in time. For example in an audio work, a cymbal strike may be an event that occurs at a certain time in an audio work. In a video work, an event may be a scene change where the pixels between frames change substantially; a blank frame preceding and/or following a non-blank frame; a frame having pixels that have one color distribution preceding or following a frame having a different color distribution; a short series of frames which begin with pixels having one color distribution and ends with pixels having a substantially different color distribution. Furthermore, for purposes of this discussion, a metric is a measurement between events. Some examples include time between events, number of frames between events, or any other easily measured attribute of the works being compared. For purposes of comparing video works, the time between events has been found to be the most dependable metric. It has been found that the time between events, which in the case of video scene changes is also known as the scene length, does not change as the format of the work is changed. In addition, the ratio between neighboring scene lengths does not change as the playback rate is changed. While a method in accordance with this invention is particularly reliable for identifying video works, one skilled in the art will recognize that this system may be used to identify other types of works and event streams as well.

[0033] This invention relates to identifying works from digital data representing the content of the work. For purposes of this discussion, content of a work is the data that actually is the representation of the work and excludes metadata, file headers or trailers or any other identification information that may be added to digital representations of the works to identify the work.

[0034] Figure 1 illustrates a network 100 that may include digital processing systems that execute instructions for performing the processes for identifying a work in accordance with this invention. Network 100 may be an Internet or Intranet connecting digital processing systems to allow the systems to transfer data between one another. One skilled in the art will recognize that a system of identifying an unknown work may be stored and executed in any processing system in the network. Furthermore, the shown components are merely examples of digital systems in a network and the exact configuration of a network is left to those skilled in the art.

[0035] In network 100, router 105 connects to network 100 via path 104. Router 105 is a conventional router that connects multiple processing systems to a network and handles data transfers to the connected systems. Path 104 is a T-1 line cable, fiber optic or other connection to another system in network 100. Desktop computer 110 and server 115 connect to router 105 via paths 111, and 114 respectively. Server 115 is a conventional server that may store data to provide data and applications to other systems such as desktop 110 connected to a local network. Typically, path 114 is an Ethernet, cable or other connection to router 105. Desktop 110 is a conventional personal computer connected to network 100. Path 111 may be an Ethernet, cable, Radio Frequency or the connection that allows communication between desktop 110 and router 105. One skilled in the art will recognize that more than two systems may be connected to router 105. The number of connections is only limited by the capacity of router 105.

[0036] Server 120 connects to network 100 via path 119. Server 120 is a conventional server that has multiple other systems connected to the server and provides network access to the connected systems. Path 119 is a T-1, telephonic, fiber optic or other type of connection between server 120 and another system in network 100. Notebook 12 and desktop computer 130 connect to server 120 via paths 124 and 129 respectively. Notebook computer 125 and desktop computer 130 are conventional personal computer systems. Paths 124 and 129 may be an Ethernet, cable, Radio Frequency or the connection that allows communication between the systems and server 120.

[0037] Router 140 connects to network 100 via path 139. Router 140 is a conventional router that connects multiple processing systems to a network and handles data transfers to the connected systems. Path 139 is T-1 line cable, fiber optic or other connection to another system in network 100. Server 145 and 150 connect to router 140 via paths 144 and 149 respectively. Servers 145 and 150 are typical servers that provide contents such as web site or other web accessible files to other users over the network. Typically, paths 144 and 149 are an Ethernet, cable or other connection to router 140.

[0038] Server 135 is also a server that provides content to users over network 100. Server 135 connects to at least one other system in network 100 via path 134. Path 134 is a T-1 line cable, fiber optic or other connection to another system in network 100. One skilled in the art will recognize that these are merely exemplary connections and the

exact configurations are left to the network administrator as being outside the scope of this invention.

[0039] Figure 2 illustrates a block diagram of the basic components of a processing system that can execute the applications to provide an identification system in accordance with this invention. One skilled in the art will recognize that this is merely an exemplary system and that the exact configuration of each processing system may be different in accordance with the requirements for the system.

[0040] Processing system 200 includes a Central Processing Unit (CPU) 201. CPU 201 is a processor, microprocessor, or a group of a combination of processors and/or microprocessors. Each processor and/or microprocessor includes registers and other circuitry for executing instructions stored in a memory to provide applications for processing data. The CPU 201 may also include firmware, which is circuitry that stores instructions for various applications.

[0041] Memory bus 205 connects CPU 201 to memories for storing executable instructions and data for applications being executed by CPU 201. A non-volatile memory such as Read Only Memory (ROM) 210 may be connected to memory bus 205. ROM 210 stores instructions for drivers and configuration data for processing system 200. A volatile memory, such as Random Access Memory (RAM) 215 is also connected to memory bus 205. RAM 215 stores data and instructions for applications being executed by CPU 201. One skilled in the art will recognize that other types of volatile memory SRAM and DRAM may also be connected. One skilled in the art will also recognize that memory caches may also be included in the memories and CPU modules.

[0042] Input/Output bus 220 connects CPU 201 to peripheral devices for transmission of data between CPU 201 and the peripheral devices. Examples of peripheral devices that may be connected to I/O bus 220 include memory 225, keyboard 230, pointing device 235, display 240, modem 245, and network connector 250. Those skilled in the art will recognize that these devices are shown for exemplary purposes and any of the devices may not be included in a processing system or other device may be included.

[0043] Memory 225 is a device for storing data and instructions for applications on a media. Memory 225 may include a disk drive for reading and writing data to a magnetic media, or an optical device for reading and/or writing data to in an optical

format to an optical media such as a compact disc. Keyboard 230 is a device receiving alphanumeric data from a user. Pointing device 235 is a mouse; touch pad or other such device used to receive input for moving an icon or "pointer" across a display. Display 240 is a device that receives data from the processing unit and displays the data on a monitor. Modem 245 is a device that connects to a telephone and converts digital data to analog signals for transmission over the telephone line. Network device 250 is a device that connects system 200 to a network to send and receive data over the network. An example of a network device 250 is an "Ethernet Card" which includes circuitry for connecting to a network.

[0044] Figures 3-8 are flow diagrams of a process for identifying a work in accordance with this invention. These processes are embodied as instructions in hardware, software and/or firmware. The instructions are then executed by a digital processing system to provide the processes as shown in the flow diagrams. One skilled in the art will recognize that any processing system may use the following processes with minor changes to the steps described. Some envisioned uses include, but are not limited to, placing the system on a router to prevent end users connected to the router from transmitting or receiving unauthorized copies of a copyrighted work; placing the system on a server that provides user downloaded content over the network to ensure that unauthorized copies of copyrighted works are not provided by the server; placing the system in an Internet browser to prevent unauthorized transfers of copyrighted works; placing the system in peer to peer software to prevent unauthorized transfers of copyrighted works; and as an utility application on a personal computer to prevent unauthorized transfers of copyrighted works.

[0045] Figures 3-6 provide flow diagrams of four exemplary embodiments of a system in accordance with this invention. One skilled in the art will recognize that individual features of any of the four embodiments may be combined in a system that operates in accordance with this invention.

[0046] Figure 3 illustrates a first exemplary process 300 for providing an identification system in accordance with this invention. Process 300 begins in step 305 when the process receives data for content of an unknown digital work. The data may be received in any number of ways including but not limited to reading the data from a medium, reading the data from memory, or extracting the data from packets being

transmitted over the network. In the preferred exemplary embodiment, the unknown work is a video work. In order to have the best chance of successfully determining the identity of the work, the system requires enough data to provide a sufficient amount of events in the video from the data. For other forms of works differing amounts of data may be needed.

[0047] In step 310, a portion of the received data is selected. This is an optional step, but, for example, if the user of this invention desires to determine whether this unknown data matches in its entirety some portion of one of the known references, said portion could be the entire received data. As another example, if the user of the invention desires to determine whether any arbitrary portion of the unknown work matches some portion of one of the references, they could specify a region of the unknown work to be examined. By calling process 300 repeatedly, a user could determine if any arbitrary portion of the received data is contained in the known reference. This could be useful if the received data were an edited collection or collage of known material. Also, the computational cost of the process can be lessened if one looks only at a portion of the received data. In step 315, the monitored events are detected in the portion of data. When applied to video works typically scene changes are the event detected. However, other events may be used such as frames in which one color is predominate.

[0048] Next, a metric between each successive event is determined in step 320. When applied to video works time between events is typically the most successful metric. However, other metrics such as numbers of frames or other attributes may be used. In step 325 a list of events as well as time from the last event is generated. In step 327, a set of known works is generated which are likely to contain the unknown work. This may be as simple as selecting all the known works in a large database or may be more sophisticated. For example, if it is known a priori that the unknown work is a television episode, said set of known works could consist only of all available television episodes. Limiting the number of known works reduces the computational cost. Also, these known works would typically be analyzed and reduced to event lists beforehand and stored in a database.

[0049] In step 330, an iterative process begins by comparing the list of events and metrics of the unknown work to a list of metrics of a known reference work. A more complete description of a preferable comparing process is given below in figures 7

through 12. The list of events metrics for each known work is stored in a database or other memory for use. The process then determines if there is a match between the lists. If there is not a match, process 300 determines whether there is another identified work to test. If so, the process is repeated from step 330. If there is not another identified work to test, step 345 determines whether there is more data of the unidentified work to test. If there is more data, the process repeats from step 310. Otherwise there is no match and a report showing the work is unknown is generated in step 347 and process 300 ends.

[0050] If there is a match between the identified work and the unknown work in step 335, the unknown work is reported as identified as the known work in step 350. In step 355, process 300 determines whether the newly identified work is an authorized copy. This may be done in any number of manners depending on how the data for the work was received. For example if the data was read from packets, the source and destination addresses may be used. The mode of delivery may also be used or any other number of methods may be used. If it is determined that the work is an authorized copy, process 300 ends. Otherwise, a report of the unauthorized copy is reported in step 360 and in an optional step 365 a business rule may be applied to the unauthorized copy. Some examples of actions that may be taken include erasing the work from the memory, blocking transmission of packets carrying the data, and degrading the data to make the unauthorized copy unusable. Process 300 then ends.

[0051] Figure 4 illustrates process 400 that is a second exemplary embodiment of this invention. Process 400 has substantially the same steps as process 300. However process 400 executes the following steps to see if a match can be made. In response to there not being another identified work in step 340, process 400 performs a secondary test on the selected portion data in step 450. In some embodiments, the test may be using the method described in the Blum patent to compare the audio portion of a work with the audio portions of the identified works.

[0052] If there is no match in the secondary test, process 400 repeats the process from step 345 as described above. If there is a match between the identified work and the unknown work in step 450 the unknown work is reported as identified as the known work in step 455. In step 460, process 400 determines whether the newly identified work is an authorized copy. This may be done by any number of methods depending on how the data for the work was received. For example if the data was read from packets, the

source and destination addresses may be used. The mode of delivery may also be used or any other number of methods may be used. If it is determined that the work is an authorized copy, process 400 ends. Otherwise, a report of the unauthorized copy is reported in step 465 and in an optional step 470 a business rule may be applied to the unauthorized copy. Some examples of actions that may be taken include erasing the work from the memory, blocking transmission of packets carrying the data, and degrading the data to make the unauthorized copy unusable. Process 400 then ends.

[0053] Figure 5 illustrates process 500 that is a third exemplary embodiment. Process 500 has the same steps as process 300 until step 350. If there is a match, a secondary test is made to confirm the identity of the unknown work.

[0054] After step 550, process 500 performs a secondary test on the selected portion of data in step 550. In some embodiments, the test may use the method described in the Blum patent to compare the audio portion of a work with the audio portions of the identified works.

[0055] If there is not a match in the secondary test, process 500 performs step 340 and tries to find another match, as the identity could not be confirmed. If there is a match between the identified work and the unknown work in step 450 the unknown work is reported as identified as the known work in step 565. In step 570, process 500 determines whether the newly identified work is an authorized copy. This may be done in any number of manners depending on how the data for the work was received. For example if the data was read from packets, the source and destination addresses may be used. The mode of delivery may also be used or any other number of methods may be used. If it is determined that the work is an authorized copy, process 500 ends. Otherwise, a report of the unauthorized copy is reported in step 575 and in an optional step 580 a business rule may be applied to the unauthorized copy. Some examples of actions that may be taken include erasing the work from the memory, blocking transmission of packets carrying the data, and degrading the data to make the unauthorized copy unusable. Process 500 then ends.

[0056] Figure 6 illustrates a fourth exemplary process 600. Process 600 provides a secondary test when the matching with an identified work in step 335 is inconclusive. An inconclusive result can happen for a number of reasons. For example, the number of elements in the unknown list of events and metrics may not be enough to say with high

likelihood that a match has been made. As another example, there may also be several known reference lists that match the unknown lists and we may need an additional test to distinguish between them. Process 600 has the following additional steps when a test is inconclusive in step 335.

[0057] In response to inconclusive results process 600 performs a secondary test on the selected portion of data and the identified work in question in step 650. In some embodiments, the test may be using the method described in the Blum patent to compare the audio portion of a work with the audio portions of the identified works.

[0058] If there is no match in the secondary test, process 600 repeats the process from step 345 as described above. If there is a match between the identified work and the unknown work in step 650 the unknown work is reported as identified as the known work in step 655. In step 660, process 600 determines whether the newly identified work is an authorized copy. This may be done by any number of methods depending on how the data for the work was received. For example if the data was read from packets, the source and destination addresses may be used. The mode of delivery may also be used or any other number of methods may be used. If it is determined that the work is an authorized copy, process 600 ends. Otherwise, a report of the unauthorized copy is reported in step 665 and in an optional step 670 a business rule may be applied to the unauthorized copy. Some examples of actions that may be taken include erasing the work from the memory, blocking transmission of packets carrying the data, and degrading the data to make the unauthorized copy unusable. Process 600 then ends.

[0059] Figures 7-11 are exemplary embodiments of a process for comparing the list of events and metrics of an unknown work against the list of events and metrics of a known work. These figures taken together correspond to step 330 in figures 3-6.

[0060] Figure 7 shows the outer loops of recursive process 700. Process 700 begins in step 705. In step 705, process 700 receives lists of events and metrics for a known work and an unknown work. In a typical embodiment in a video matching application, each list includes the time locations of scene changes of the video and other metrics such as the strength of the scene change, e.g., the average frame-to-frame pixel difference normalized by the average luminance of the neighboring frames. In step 710, process 700 sets an integer index variable N to 1 to initialize a recursive loop. N is an index over the events and metrics in the list of the known and unknown works. The outer

loop, indexed by N, is used to align and test the matching of the unknown at each of the event locations in the known reference. The loop begins in step 715. In step 715, process 700 determines whether N is greater than or equal to the length of the list for the known work. If N is greater than the list length, process 700 proceeds to step 760 and ends. If N is not greater than or equal to the length of the list for the known work, process 700 proceeds to step 720. In step 720, the process sets a second integer index variable, M to 1. This second integer index variable is an index over the first few elements of the list of events and metrics for the unknown work. The inner loop determines whether the first event in the unknown list has a corresponding matching event in the known list. If the first event does not have a match in the known list, a match may not be detected even in a case where all the following events in the unknown matched events and metrics in the known work match perfectly.

[0061] In process 700, step 725 tests the index M against the minimum of 3 and the length of the unknown list. The inner loop of process 700 tests the basic matching algorithm by aligning each of the first 3 events in the unknown list with event N in the known list. Anyone skilled in the art will realize that a more exhaustive search can be conducted by raising the constant in step 725. If M is greater than this minimum, process 700 increments the outer index N in step 730 and repeats process 700 from step 715.

[0062] If the test fails, process 700 proceeds to step 735, where the Mth element of the unknown work list is aligned with the Nth element of the known work list. In step 740, process 700 performs a low-level comparison to detect a match between the Mth element of the unknown work list aligned with the Nth element of the known work list. Figures 8 and 9 show a process for the comparison test of step 740. Process 700 enters step 740 with a) a list of events and metrics from the unknown work list starting at event M and continuing to the end of the list of events and metrics in the unknown; b) a list of events and metrics from the known work list starting at event N and continuing to the end of the list of events and metrics in the known; c) the number of hits set to 1; and d) the number of misses set to 0. At 745 we test the results of 740. If there is no match reported, process 700 increments M in step 750 and return to the top of the inner loop at 725. If there is a match reported, process 700 adds this match to the list of matches and increments N at 730 and proceeds to repeat process 700 from step 715. In an alternate

embodiment, process 700 will return after the first match rather than look for all possible matches.

[0063] Figure 8 shows an exemplary embodiment of the low-level comparison process 800 which performs a low level comparison for step 740. At the beginning of the process 800, process 800 aligns one of the events in the unknown list with an event in the known reference list. Process 800 measures how well the following events and metrics in the two lists correspond. A high degree of correspondence will result in a match being reported and a low degree of correspondence will result in a report of no match. In the following description, a low-level "miss" occurs when there is either no event in the known list corresponding to a particular event in the unknown list or there is no event in the unknown list corresponding to a particular event in the known reference list. A low-level "hit" occurs when there is a correspondence between an element in the known reference list and an event in the unknown list. The process shown in figure 8 is a recursive process. Process 800 causes another recursion of process 800 from various locations inside process 800, and a reported match or no match deep inside the recursion will be returned up to the top of the recursion.

[0064] Process 800 begins in step 805 by comparing a number of total misses so far against a miss threshold. The miss threshold typically is a relative threshold, for example, a percentage of the total number of events that have been examined by process 800. The first time process 800 enters the recursive process the number of misses is 0. If the miss threshold is exceeded by the number of misses, step 810 returns an indication of no match and returns to the next recursion of the process. If the threshold is not exceeded, process 800 performs step 815.

[0065] In step 815, process 800 determines if the end of the unknown list has been reached in the current recursion of process 800. If the end of the unknown list has been reached, process 800 performs an evaluation step 820 and reports the result of the evaluation in step 825 and returns. This evaluation step will be described more in the following figures. In a typical embodiment, evaluation 820 uses the number of misses, the number of hits, some scoring or error measure of the hits, and a weighting of hits and misses and error based on the other metrics of the events, e.g., how strong of an event it is.

[0066] In step 822, process 800 determines whether the end of the known list has been reached. If the end of known list has been reached process 800 returns no match in step 867 and returns to the prior recursion of process 800 in step 870.

[0067] If this recursion of process 800 has not reached the end of the unknown list, process 800 considers the next element of the unknown list in step 830. Process 800 generates a search list of elements in the known list in step 835. The construction of the error bound is shown in more detail in figures 10 and 11. The search list contains all the events in the known reference list that are within a certain distance of the current event in the aligned unknown.

[0068] Step 845 determines if the search list is empty. If the search list is empty, process 800 updates the number of misses. For purposes of this disclosure, an empty search list has no events in the unknown list that are close in metrics to the current event in the known list. Process 800 updates the number of misses in the known miss total and adds one to the number of misses in the unknown miss total; updates a list showing all the correspondences of events so far; moves an index to the next element in the unknown; and calls for a recursion of process 800 from the current element of the unknown list. The number of misses in the known miss total is incremented by the number of events in the known event list that are skipped over before the next event in the known list is found that is associated with the said unknown element.

[0069] If the search list is not empty, step 850 sets a third index integer, I, to 0. In step 860, process 800 determines whether I is greater than the length of the search list. If not, process 800 increments the third index integer variable in step 865 and step 860 is repeated. Steps 850-865 associate the next element in the unknown list considered in step 830 with each element of the search list, and call the process in 800 recursively for each iteration of I. Each recursive call has to keep track of all past associations of unknown events and known reference events as well as current miss and hit counts so that the tests in 805 and 820 can be applied.

[0070] At some point in the recursive calling sequence, process 800 reaches the end of the unknown list in step 815 and the current recursion ends. Each recursion of process 800 returns until a match or no match is reported to the test in step 745 of figure 7.

[0071] Figure 9 is an exemplary process 900 for performing the evaluation step 820 of process 800 shown in figure 8. In a typical embodiment, process 900 takes into account the number of misses, the number of hits, some scoring or error measure of the hits, and a weighting of hits and misses and error based on the other metrics of the events, e.g., the strength of an event. At this point the process is at the end of the unknown and will make a final decision as to whether the candidate list of associations of events in the unknown and events in the known reference are indicative of a match. Process 900 begins in step 910 by receiving the list of associated pairs of events in the unknown list and events in the known list as well as the accumulated number of hits and misses. In step 920, process 900 computes a regression line through the set of points (x,y) where each x is the time location of an element of the unknown list which has an associated element in the unknown list and y is the time location of said associated known list element. The regression line can be computed by standard linear regression techniques well known in the art.

[0072] After the regression line is computed, process 900 computes a regression error in 930. In step 930, process 900 computes an unweighted regression error and a regression error weighted by the strength of the event from the unknown list or other metrics associated with the event. In an exemplary embodiment, the regression error will be computed by taking the average of the absolute error between the timing of each event in the unknown list and the timing of the associated event in the known list. The timing of the unknown event has to be measured relative to the first event in the unknown list and the timing of the event from the known list has to be measured relative to the known event associated with the first event in the unknown list. The timing of the known events has to be corrected by the slope and offset of the regression line. The weighted regression error is computed in the same manner: a weighted average is computed, where the weight at each step is the strength of the event or another metric associated with the event. Also in step 930, process 900 computes the ratios of misses to the sum of the number of hits and misses in both a weighted and unweighted version, where the weights are the strength of the event or another metric associated with the event.

[0073] In step 940, process 900 tests each of these errors and ratios against an associated threshold to determine if the match is within the error limits. In an exemplary embodiment threshold1, a regression error limit, is set to 0.05; threshold2, a weighted

regression error limit, is set to 0.04; threshold3, a miss ratio limit, is 0.3; and threshold4, a weighted ratio limit, is 0.25. Furthermore, in step 930, process 900 also compares the total number of hits to a minimum number of hits to avoid degenerate cases. In an exemplary embodiment, the minimum number of hits is set to 10. Based on this test, process 900 either reports a match in 960 or reports no match in 950 and process 900 returns to the recursion in step 970.

[0074] Figures 10 and 11 show an exemplary method for setting the error bounds in step 835 in figure 8. Figure 10 shows how the bound is calculated the first time through step 835 and figure 11 shows a tighter bound that can be applied for later iterations. In figure 10, the x axis holds the event locations of the unknown and the y axis holds the event locations of the reference signature. Without loss of generality, the figure 10 shows the x(0),y(0) pair at the origin for convenience. This is the location of the test alignment between the two lists resulting from step 735. If the two lists were identical at this point, the associated x,y pairs that follow would all lie along the line x=y with a slope of 1. However, the fact that the playback rate can be off by an amount R means that the points can fall within a cone from a slope of 1-R to a slope of 1+R. In an exemplary embodiment, R is set to 5% (0.05) which covers typical rate changes in video due to NTSC to PAL conversion, for example. In addition, process 900 may allow for a "jitter error" J on the alignment between corresponding points due to errors caused for example by frame rate changes and scene change detection. In an exemplary embodiment, J is set to 0.1 seconds, which covers typical frame rate changes in video on the Internet, for example. That is, the initial x(0),y(0) pair can be off by +-J, which is shown by the red sloping lines above and below the cone. In addition, the final x(n),y(n) pair can be off by +-J, which is shown by the +-J bars along the y axis. The final error bounds are shown on the right, from ((1-R)x(n) - 2J) to ((1+R)x(n) + 2J). The search list in 835 will consist of all the elements in the known reference list whose event timings fall between these bounds.

[0075] Figure 11 shows that once at least a pair of associated events are found, previous associations of event timings constrain the possibilities for new associations events further along the known and unknown lists. In this case, process 800 proceeded far enough such that a list of associated points from x(0),y(0) to x(n),y(n) have been

tested and process 800 is now searching for a possible match with the timing x(n+1) of the next element in the unknown list.

[0076] Both the beginning and ending pairs can be misaligned by +/-J, which gives two worst-case lines, one from the maximum positive misalignment of the first pair through the maximum negative misalignment of the last pair and the other from the maximum negative misalignment of the first pair through the maximum positive misalignment of the last pair. Process 800 also allows for a +-J misalignment error at the final choice of y(n+1). This gives the bounds on the right, from ((y(n)-2J)x(n+1)/x(n) - J) to ((y(n)+2J)x(n+1)/x(n) + J). The search list in 835 now includes all the elements in the known reference list whose event timings fall between the tighter of these bounds and the bounds determined by the calculation shown in figure 10.

[0077] Referring back to Figure 7, the outer loop of process 700 surrounded by steps 710, 715 and 730 of figure 7 exemplifies a 'brute force' search where every possible alignment of the unknown reference list against the known reference list is tested. In general, a brute force method is inefficient. Thus, various methods can be used to index the list of the known work to speed up the search process. In a typical embodiment of this invention, a database contains all the known reference lists for all the target videos or media files of interest. Before any unknown videos or files are processed, the database is indexed to speed up the search. As an example, each event in the event list for each known work would be associated with the time duration between said event timing and the timing of the next event in the said list. A sorted list of all the durations for all the events in all the event lists for all the known references would be stored using any sorting technique well known in the art. Each of these durations would have an associated pointer referring back to the events associated with the duration.

[0078] When such a list for an unknown work is passed to search process 700, the outer loop is not required and may be removed. Figure 12 illustrates a process that is the equivalent of process 700 using this index. Most of process 1200 is similar to process 700, except that the outer loop of 700 has been removed and replaced by a new inner loop that uses the index to choose particular locations in the known reference list instead of brute force searching over all locations in the said list.

[0079] In step 1220, process 1200 computes the duration between the Mth event in the unknown list and M+1th event in the unknown list. Using any sorted list search

technique such as binary search or hash search, process 1200 generates a list of events in the known reference list in step 1225 which have durations close to the duration computed in step 1220. For purposes of this discussion, 'close to' means durations within an error bound determined by the expected jitter error J and the playback rate error R.

[0080] The inner loop specified by steps 1230, 1235 and 1245 iterates over all the elements in this search list. The low-level comparison in step 1225 is exactly the same as that in step 740 of figure 7. One skilled in the art will recognize that other attributes of the list of events and metrics could be used to limit the search to those Mth events that are most likely to match. Some examples include using strength of an event within some tolerance, the absolute value of the measured signal slightly before or slightly after the event, or, if the signal is multidimensional, then some distribution of the signal slightly before or after the event could be used.

[0081] The above is a description of various embodiments in accordance with this invention. It is anticipated that those skilled in the art can and will design alternative systems that infringe on this invention as set forth in the following claims.

CLAIMS

1. A method for identifying a work in a digital processing system comprising:

selecting a portion of data of an unknown work;

detecting each event in said portion of data of said unknown work, wherein events are perceptual occurrences in a work that can be successively positioned in time;

determining an event metric between each successive event in said portion of data in said unknown work;

generating a first list of event metrics between said events for said unknown work;

receiving a second list of event metrics for a known work;

generating a third list comprising events in the first list that match events in the second list by

performing a comparison to determine whether an Mth event of the first list matches an Nth event of the second list, wherein M is an index over events in the first list and N is an index over events in the second list;

responsive to a determination of a match, adding the matching events to the third list and incrementing the index N;

responsive to a determination that there is not a match, incrementing the index M,

wherein the comparison is performed until the index N is greater than or equal to a number of events in the second list and the index M is greater than a threshold; and

determining said unknown work is a copy of said known work based on the third list comprising the events in the first list that match events in the second list.

2. The method of claim 1 wherein the threshold is a minimum of 3 and a length of the second list.

3. The method of claim 1 wherein determining said unknown work is a copy of said known work comprises:

generating the third list wherein each M+X event of said first list for said unknown work is paired with the N+X event of said second list for said known work; and

computing a regression line through said associated pairs in said list of associated pairs.

4. The method of claim 3 further comprising:

computing an error tolerance from said regression line.

5. The method of claim 3 further comprising:

determining a regression error from said computation of said regression line;

comparing said regression error to a regression error threshold; and

determining said unknown work is a copy of said known work responsive to said regression error being less than said regression error threshold.

6. The method of claim 3 further comprising:

determining a weighted error from said computation of said regression line;

comparing said weighted error to a weighted error threshold; and

determining said unknown work is a copy of said known work responsive to said weighted error being less than said weighted error threshold.

7. The method of claim 3 further comprising:

determining a miss ratio from said computation of said regression line;

comparing said miss ratio to a miss ratio threshold; and

determining said unknown work is a copy of said known work responsive to said miss ratio being less than said miss ratio threshold.

8. The method of claim 3 further comprising:

determining a weighted miss ratio from said computation of said regression line;

comparing said weighted miss ratio to a weighted miss ratio threshold; and

determining said unknown work is a copy of said known work responsive to said weighted miss ratio being less than said weighted miss ratio threshold.

9. A system for identifying a digital work, the system being configured to:

select a portion of data of an unknown work;

detect each event in said portion of data of said unknown work, wherein events are perceptual occurrences in a work that can be successively positioned in time;

determine an event metric between each successive event in said portion of data in said unknown work;

generate a first list of event metrics between said events for said unknown work;

receive a second list of event metrics for a known work;

generate a third list comprising events in the first list that match events in the second list; and

determine said unknown work is a copy of said known work based on the third list comprising the events in the first list that match events in the second list, wherein the system is configured to generate the third list by

performing a comparison to determine whether an Mth event of the first list matches an Nth event of the second list, wherein M is an index over events in the first list and N is an index over events in the second list, adding the matching events to the third list and incrementing the index N responsive to a determination of a match, and responsive to a determination that there is not a match, incrementing the index M, wherein the comparison is performed until the index N is no longer less than a number of events in the second list and the index M is greater than a threshold.

10. The system of claim 9 wherein the threshold is a minimum of 3 and a length

of the second list.

11. The system of claim 9 wherein determining said unknown work is a copy of said known work comprises the system

generating the third list wherein each M+X event of said first list for said unknown work is paired with the N+X event of said second list for said known work; and

computing a regression line through said associated pairs in said list of associated pairs.

12. The system of claim 11 further configured to:

compute an error tolerance from said regression line.

13. The system of claim 11 further configured to:

determine a regression error from said computation of said regression line;

compare said regression error to a regression error threshold; and

determine said unknown work is a copy of said known work responsive to said regression error being less than said regression error threshold.

14. The system of claim 11 further configured to:

determine a weighted error from said computation of said regression line;

compare said weighted error to a weighted error threshold; and

determine said unknown work is a copy of said known work responsive to said weighted error being less than said weighted error threshold.

15. The system of claim 11 further configured to:

determine a miss ratio from said computation of said regression line;

compare said miss ratio to a miss ratio threshold; and

determine said unknown work is a copy of said known work responsive to said miss ratio being less than said miss ratio threshold.

16. The system of claim 11 further configured to:

determine a weighted miss ratio from said computation of said regression line;

compare said weighted miss ratio to a weighted miss ratio threshold; and

determine said unknown work is a copy of said known work responsive to said weighted miss ratio being less than said weighted miss ratio threshold.

17. A computer program product comprising instructions which, when implemented on a computer, cause it to carry out a method in accordance with any of claims 1 to 8.