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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
7 January 2016 (07.01.2016)

WIPO | PCT

(10) International Publication Number
WO 2016/003436 A1

(51) International Patent Classification:
H04N 7/15 (2006.01)

(21) International Application Number:
PCT/US2014/044969

(22) International Filing Date:
30 June 2014 (30.06.2014)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.** [US/US]; 11445 Compaq Center Drive West, Houston, Texas 77070 (US).

(72) Inventors: **SO, Chi**; 11445 Compaq Center Dr W, Houston, Texas 77070 (US). **BIGGS, Kent E.**; 11445 Compaq Center Dr W, Houston, Texas 77070 (US). **STEVENS, Jeffrey C.**; 11445 Compaq Center Dr W, Houston, Texas 77070 (US).

(74) Agents: **POJUNAS, Scott A.** et al.; Hewlett-Packard Company, Intellectual Property Administration, 3404 E. Harmony Road, Mail Stop 35, Fort Collins, Colorado 80528 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *as to the identity of the inventor (Rule 4.17(i))*

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

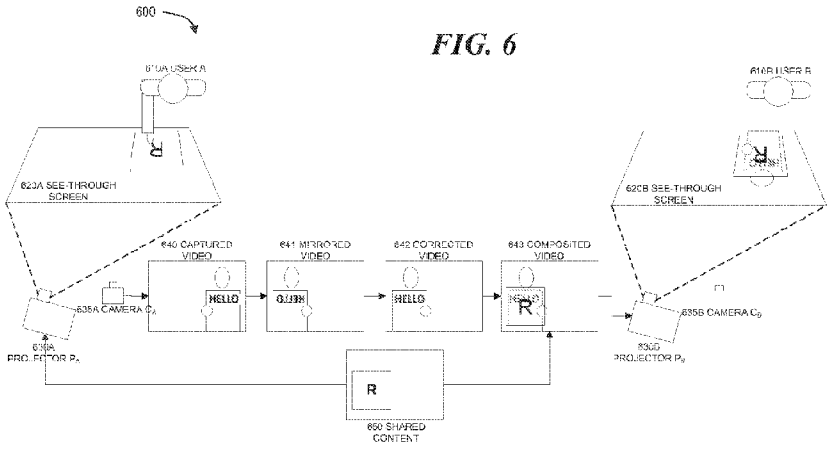
Published:

— *with international search report (Art. 21(3))*

WO 2016/003436 A1

(54) Title: CHARACTER RECOGNITION IN REAL-TIME VIDEO STREAMS

FIG. 6



(57) Abstract: Examples relate to detecting character data. At least one shape and characteristics related to the at least one shape are stored in a database. When a video stream is obtained, it may be determined whether the video stream comprises video content showing a shape having the characteristics of the at least one shape. In response to determining that the video stream comprises video content showing a shape having the characteristics of the at least one shape, the character data within that shape may be detected.

CHARACTER RECOGNITION IN REAL-TIME VIDEO STREAMS

BACKGROUND

[0001] Remote collaboration systems strive to deliver an experience that local and remote meeting participants are in the same room. In particular, a see-through screen-based collaboration system creates the illusion that the users are seemingly separated only by a sheet of glass where in fact they are at different locations. Such a system provides an effective collaboration tool where the users can see each other's body language, hand gestures, eye contact, and gaze.

BRIEF DESCRIPTION OF THE DRAWINGS

[0002] The following detailed description references the drawings, wherein:

[0003] FIG. 1 is an example environment in which various examples may be implemented as a video processing system.

[0004] FIG. 2 is a block diagram depicting an example machine-readable storage medium comprising instructions executable by a processor for video processing.

[0005] FIG. 3 is a flow diagram depicting an example method for detecting character data in a video stream based on known shapes.

[0006] FIG. 4 is a flow diagram depicting an example method for video processing as used in a see-through screen-based collaboration system.

[0007] FIG. 5 is an example picture depicting how two users communicate using a see-through screen-based collaboration system.

[0008] FIG. 6 is a diagram depicting an example system design of a see-through screen-based collaboration system.

[0009] FIG. 7 is a diagram depicting an example implementation of detecting character data in a video stream.

[0010] FIG. 8 is a diagram depicting an example implementation of controlling a mirror image effect using a depth camera.

DETAILED DESCRIPTION

[0011] The following detailed description refers to the accompanying drawings. Wherever possible, the same reference numbers are used in the drawings and the following description to refer to the same or similar parts. It is to be expressly understood, however, that the drawings are for the purpose of illustration and description only and are not intended as a definition of the limits of the invention. While several examples are described in this document, modifications, adaptations, and other implementations are possible. Accordingly, the following detailed description does not limit the disclosed examples. Instead, the proper scope of the disclosed examples may be defined by the appended claims.

[0012] Remote collaboration systems strive to deliver an experience that local and remote meeting participants are in the same room. In particular, a see-through screen-based collaboration system can create the illusion that the users are seemingly separated only by a sheet of glass where in fact they are physically at different locations. Such a system provides an effective collaboration tool where the users can see each other's body language, hand gestures, eye contact, gaze, and how they interact with shared content displayed on the see-through screen.

[0013] In some implementations, the see-through screen-based collaboration system may include a first video capturing device (e.g., video camera) that captures a view of a first user (as well as the space around the first user) through a first see-through screen. The "see-through screen" as used herein may comprise a transparent display screen through which a user can view, upload, or otherwise interact with content (e.g., image, text, video, etc.) and that the user can write on. An

example see-through screen is shown in FIG. 5. The first video capturing device may be installed behind the first screen, shooting through the first screen. The first user may be present on the other side of the screen, facing the first camera through the first screen. A first projector may be installed on the same side as the first camera, projecting shared content on the first screen.

[0014] A similar arrangement of the system may be set up at a remote location for a second user. For example, a second video capturing device may capture a view of the second user (as well as the space around the second user) through a second see-through screen. The second video capturing device may be installed behind the second screen, shooting through the second screen. The second user may be present on the other side of the screen, facing the second camera through the second screen. The content being shared and interacted with by the first user as well as the captured view of the first user may be projected by a second projector to the second see-through screen. Likewise, the content being shared and interacted with by the second user as well as the captured view of the second user may be projected by the first projector to the first see-through screen. This particular arrangement of the see-through screen-based system would allow for capturing a video image of the remote user from a viewpoint that corresponds to that of the local user. An example system design of the see-through screen-based collaboration system as discussed above is illustrated in FIG. 6.

[0015] The "shared content" as used herein may comprise any content (e.g., image, text, video, etc.) that may be shared by a user via his/her see-through screen. The shared content may then be projected to and/or displayed on another user's see-through screen. In this manner, the local and remote users may view the same content on their respective see-through screens as the content is being created, uploaded, interacted with, and/or manipulated by the users. For example, the first user may draw a flow diagram on the first see-through screen. The user may also upload and/or share content on the first screen or may choose to draw on top of an uploaded image. All of these may be projected to and/or displayed on the second

see-through screen via which the second user may view the shared content. The second user may further manipulate the shared content and any changes and additions made to the shared content may be simultaneously displayed on the first see-through screen.

[0016] Note that any shared content between the first and second users may be displayed in the same orientation so that drawings (or any other content) on the screen appear correct to both users. Assuming that two users are physically located in the same room and are separated by a transparent glass, one user may write the word "logistics" on his/her side of the transparent glass. That word will appear mirror-reversed on the other side of the glass, making it very difficult for the user on the other side to read or understand. Thus, in the see-through screen-based collaboration system as discussed herein, any shared content may be displayed on the screen in such a way that both users will be able to see the word "logistics" in its correct orientation rather than a mirror-reversed version of the word.

[0017] Because the shared content will be displayed in the same orientation for both users, the captured video images of the remote user may be horizontally flipped (or mirror-reversed) so that the shared content and the captured video images of the remote user may be properly combined and displayed on the local user's see-through screen. For example, a video image of the first user may be mirror-reversed and displayed on the second see-through screen. Likewise, a video image of the second user may be mirror-reversed and displayed on the first see-through screen. However, just like looking at text in a mirror, text shown in the mirrored video image will appear to be written backwards. For example, when the first user is wearing a t-shirt with a word, that word will be shown backwards if viewed from the second user's perspective.

[0018] Examples disclosed herein address this undesirable mirror image effect by recognizing and/or detecting character data (e.g., letters, numbers, symbols, etc.) in a video stream. The character data may be recognized and/or detected based on various different ways. For example, the character data may be detected by

recognizing a shape that is known to include character data. In one example, T-shirts often have some text written on them. The mirror image effect may be corrected by horizontally flipping a portion of the video stream representing the detected character data around an axis (e.g., the central axis of the detected portion). Resolution and/or accuracy of the detected character data may be enhanced and the enhanced character data may be displayed on the screen. Further, any perspective distortion caused by the flipping of the detected portion may be corrected to deliver an even more realistic experience.

[0019] FIG. 1 is an example environment 100 in which various examples may be implemented as a video processing system 110. Environment 100 may include various components including server computing device 130 and client computing devices 140 (illustrated as 140A, 140B, ..., 140N). Each client computing device 140A, 140B, ..., 140N may communicate requests to and/or receive responses from server computing device 130. Server computing device 130 may receive and/or respond to requests from client computing devices 140. Client computing devices 140 may include any type of computing device providing a user interface through which a user can interact with a software application. For example, client computing devices 140 may include a laptop computing device, a desktop computing device, an all-in-one computing device, a tablet computing device, a mobile phone, an electronic book reader, a network-enabled appliance such as a "Smart" television, and/or other electronic device suitable for displaying a user interface and processing user interactions with the displayed interface. While server computing device 130 is depicted as a single computing device, server computing device 130 may include any number of integrated or distributed computing devices serving one or more software applications for consumption by client computing devices 140.

[0020] The various components (e.g., components 129, 130, and 140) depicted in FIG. 1 may be coupled to at least one other component via a network 50. Network 50 may comprise any infrastructure or combination of infrastructures that enable electronic communication between the components. For example, network 50 may

include any one or more of the Internet, an intranet, a PAN (Personal Area Network), a LAN (Local Area Network), a WAN (Wide Area Network), a SAN (Storage Area Network), a MAN (Metropolitan Area Network), a wireless network, a cellular communications network, a Public Switched Telephone Network, and/or other network. According to various implementations, video processing system 110 and the various components described herein may be implemented in hardware and/or programming that configure hardware. Furthermore, in FIG. 1 and other Figures described herein, different numbers of components or entities than depicted may be used.

[0021] Video processing system 110 may process a video stream captured by a video capturing device in such a way that allows the local and remote users to have a realistic collaboration experience that they are facing each other through a see-through glass while interacting with each other through shared content displayed on the see-through glass. Video processing system 110 may create a mirrored video image of the video stream captured by the video capturing device, recognize and/or detect character data in the mirrored video image, and/or correct the undesirable mirror image effect by flipping a portion in the mirrored video image representing the detected character data. Any perspective distortion caused by the flipping of the portion representing the detected character data may be corrected as well. Further, video processing system 110 may enhance resolution and accuracy of the detected character data.

[0022] To facilitate these and other functions, video processing system 110 may comprise a video mirroring engine 121, a detecting engine 122, a correcting engine 123, a character enhancing engine 124, and an output generating engine 125, and/or other engines. The term “engine”, as used herein, refers to a combination of hardware and programming that performs a designated function. As is illustrated respect to FIG. 2, the hardware of each engine, for example, may include one or both of a processor and a machine-readable storage medium, while the programming is a

set of instructions or code stored on the machine-readable storage medium and executable by the processor to perform the designated function.

[0023] Video mirroring engine 121 may create and/or generate a mirrored version of a video stream captured by a video capturing device. As discussed above, because the shared content will be displayed in the same orientation for both users, the captured video images of the remote user may be horizontally flipped (or mirror-reversed) so that the shared content displayed on the screen appear correct to both users. For example, the video content of the video stream may be flipped horizontally around the central axis of the captured video image. In this example, the video image of the first user may be mirror-reversed and projected to the second see-through screen. Likewise, the video image of the second user may be mirror-reversed and projected to the first see-through screen.

[0024] Detecting engine 122 may recognize and/or detect character data (e.g., letters, numbers, symbols, etc.) in a video stream. The character data may be recognized and/or detected based on various different ways. In one example, the character data may be detected by graphically matching the characters to known characters stored and maintained in a database (e.g., character database). In this example, the OCR (optical character recognition) technology may be used to graphically match the characters to the known characters. The character database may be updated over time to include additional characters, fonts, or glyphs for better recognition. Both English and foreign language characters along with punctuation marks or other symbols may be included in the character database.

[0025] In another example, detecting engine 122 may recognize a shape that is known to contain character data. A database (e.g., shape database) may store at least one shape that is known to contain character data and its shape characteristics. Shapes such as a square (e.g., a piece of paper, signs, posters, etc.), rectangle (e.g., a piece of paper, signs, posters, etc.), octagon (e.g., a stop sign), T-shirt shape, or a flag shape often have some letters, numbers, or symbols within the shapes. Such shapes will be scrutinized more carefully to see if they contain character data since

they are more likely to contain the character data than other types of objects or shapes. Which shapes to be included in the shape database may be determined based on user input. In some instances, a machine learning algorithm may be used to identify the shapes that have been previously determined to actually contain character data for inclusion in the shape database. In some implementations, once a particular shape having the characteristics of at least one of the shapes stored in the shape database is detected in the video stream, detecting engine 122 may detect character data within the shape.

[0026] In yet another example, a view captured by a video capturing device may have a designated section of the view where any character data within the designated section may be recognized by detecting engine 122. In this example, a user can hold a sheet of paper up so that any character data written on the paper is positioned within the designated section of the view. Detecting engine 122 may then recognize and/or detect the character data shown within the designated section of the camera view. An example diagram depicting the use of the designated section of the camera view to detect character data is illustrated in FIG. 7.

[0027] Correcting engine 123 may correct the mirror image effect by horizontally flipping a portion of the video stream representing the detected character data around an axis (e.g., the central axis of the detected portion). In some implementations, correcting engine 123 may correct the mirror image effect by flipping an object in the shape recognized by detecting engine 122. For example, when a T-shirt shape is recognized in the video stream, the shape may be flipped horizontally around the central axis of the T-shirt shape, allowing any character data within the shape to be flipped around as well.

[0028] In some implementations, the video capturing device may comprise a depth camera that can determine the distance to a human or other object in a field of view of the camera. Using the depth camera, correcting engine 123 may mirror reverse (or flip horizontally) any characters, physical objects, or even the entire space that are between a first distance from the depth camera to a second distance from the

depth camera. In some instances, the viewing space of the camera may be divided into several different depth ranges. For each depth range, correcting engine 123 may specify what needs to be mirror reversed and what needs to remain unchanged (e.g., not mirrored). In one example, correcting engine 123 may mirror reverse (or flip horizontally) only the character data that is present within 1-3 feet from the depth camera. The orientation of any other physical objects or the space itself within that depth range (e.g., 1-3 feet) may remain unchanged. Correcting engine 123 may mirror reverse (or flip horizontally) the entire space within 3-6 feet from the depth camera whereas the orientation of the space beyond 6 feet from the depth camera may remain the same.

[0029] Further, correcting engine 123 may correct any perspective distortion caused by the flipped portion of the video stream representing the detected character data. When a particular portion of the video stream is flipped, it may cause the video to have unnatural perspective distortion. For example, if the word "HELLO" is written on a piece of paper where one end of the paper is closer to the screen than the other end, then the letter "H" might look larger than the letter "O" in the captured video image. When the word is flipped while the orientation of the paper stays the same, the final video image may look unnatural and might severely compromise the realistic collaboration experience. Thus, any perspective distortion caused by the flipped character portion may be corrected to deliver an even more realistic collaboration experience.

[0030] Character enhancing engine 124 may enhance the recognized and/or detected character data in the video stream. The video images of the recognized characters may be made to be shaper, clearer, and more accurate. In one example, character enhancing engine 124 may find closest matching fonts or glyphs and replace the detected characters with those fonts or glyphs. Fonts, glyphs, and/or other related data may be stored in a fonts database and/or other databases.

[0031] Output generating engine 125 may generate an output video stream to be projected to a see-through screen. The output video stream may comprise the

mirrored video stream (by video mirroring engine 121) with the flipped portion (by correcting engine 123) and the shared content. The shared content may be combined with the mirrored version of the captured video stream (with the flipped portion as discussed herein) to create the output video stream. The output video stream may be projected to and/or displayed on a see-through screen in front of a remote user. For example, the output video stream capturing a view of the first user may be displayed on the second see-through screen. Similarly, the output video stream capturing a view of the second user may be displayed on the first see-through screen.

[0032] In performing their respective functions, engines 121-125 may access data storage 129. Data storage 129 may represent any memory accessible to video processing system 110 that can be used to store and retrieve data. Data storage 129 may comprise floppy disks, hard disks, optical disks, tapes, solid state drives, random access memory (RAM), read-only memory (ROM), electrically-erasable programmable read-only memory (EEPROM), flash drives, portable compact disks, and/or other storage media for storing computer-executable instructions and/or data. Video processing system 110 may access data storage 129 locally or remotely via network 50 or other networks. In some implementations, data storage 129 may comprise the character database, the shape database, the fonts database, and/or other database as discussed herein.

[0033] Data storage 129 may include a database to organize and store data. Database may be, include, or interface to, for example, an Oracle™ relational database sold commercially by Oracle Corporation. Other databases, such as Informix™, DB2 (Database 2) or other data storage, including file-based (e.g., comma or tab separated files), or query formats, platforms, or resources such as OLAP (On Line Analytical Processing), SQL (Structured Query Language), a SAN (storage area network), Microsoft Access™, MySQL, PostgreSQL, HSpace, Apache Cassandra, MongoDB, Apache CouchDB™, or others may also be used, incorporated, or accessed. The database may reside in a single or multiple physical device(s) and in

a single or multiple physical location(s). The database may store a plurality of types of data and/or files and associated data or file description, administrative information, or any other data.

[0034] FIG. 2 is a block diagram depicting an example machine-readable storage medium 210 comprising instructions executable by a processor for video processing.

[0035] In the foregoing discussion, engines 121-125 were described as combinations of hardware and programming. Engines 121-125 may be implemented in a number of fashions. Referring to FIG. 2, the programming may include processor executable instructions 221-225 stored on a machine-readable storage medium 210 and the hardware may include a processor 211 for executing those instructions. Thus, machine-readable storage medium 210 can be said to store program instructions or code that when executed by processor 211 implements video processing system 110 of FIG. 1.

[0036] Machine-readable storage medium 210 may be any electronic, magnetic, optical, or other physical storage device that contains or stores executable instructions. In some implementations, machine-readable storage medium 210 may be a non-transitory storage medium, where the term “non-transitory” does not encompass transitory propagating signals. Machine-readable storage medium 210 may be implemented in a single device or distributed across devices. Likewise, processor 211 may represent any number of processors capable of executing instructions stored by machine-readable storage medium 210. Processor 211 may be integrated in a single device or distributed across devices. Further, machine-readable storage medium 210 may be fully or partially integrated in the same device as processor 211, or it may be separate but accessible to that device and processor 211.

[0037] In one example, the program instructions may be part of an installation package that when installed can be executed by processor 211 to implement video processing system 110. In this case, machine-readable storage medium 210 may be a portable medium such as a floppy disk, CD, DVD, or flash drive or a memory

maintained by a server from which the installation package can be downloaded and installed. In another example, the program instructions may be part of an application or applications already installed. Here, machine-readable storage medium 210 may include a hard disk, optical disk, tapes, solid state drives, RAM, ROM, EEPROM, or the like.

[0038] Processor 211 may be one or more central processing units (CPUs), microprocessors, and/or other hardware devices suitable for retrieval and execution of instructions stored in machine-readable storage medium 210. Processor 211 may fetch, decode, and execute program instructions 221-225, and/or other instructions. As an alternative or in addition to retrieving and executing instructions, processor 211 may include one or more electronic circuits comprising a number of electronic components for performing the functionality of one or more of instructions 221-225, and/or other instructions.

[0039] In FIG. 2, the executable program instructions in machine-readable storage medium 210 are depicted as video mirroring instructions 221, detecting instructions 222, correcting instructions 223, character enhancing instructions 224, and output generating instructions 225. Instructions 221-225 represent program instructions that, when executed, cause processor 211 to implement engines 121-125, respectively.

[0040] FIG. 3 is a flow diagram depicting an example method 300 for detecting character data in a video stream based on known shapes. The various processing blocks and/or data flows depicted in FIG. 3 (and in the other drawing figures such as FIG. 4) are described in greater detail herein. The described processing blocks may be accomplished using some or all of the system components described in detail above and, in some implementations, various processing blocks may be performed in different sequences and various processing blocks may be omitted. Additional processing blocks may be performed along with some or all of the processing blocks shown in the depicted flow diagrams. Some processing blocks may be performed simultaneously. Accordingly, method 300 as illustrated (and described in greater detail below) is meant be an example and, as such, should not be viewed as limiting.

Method 300 may be implemented in the form of executable instructions stored on a machine-readable storage medium, such as storage medium 210, and/or in the form of electronic circuitry.

[0041] Method 300 may start in block 310 and proceed to block 321 where at least one shape and characteristics related to the at least one shape are stored in a database. The database (e.g., shape database) may store at least one shape that is known to contain character data and its shape characteristics. Shapes such as a square (e.g., a piece of paper, signs, posters, etc.), rectangle (e.g., a piece of paper, signs, posters, etc.), octagon (e.g., a stop sign), T-shirt shape, or a flag shape often have some letters, numbers, or symbols within the shapes. Such shapes will be scrutinized more carefully to see if they contain character data since they are more likely to contain the character data than other types of objects or shapes.

[0042] In block 322, a video stream may be obtained. In some implementations, the video stream may be a mirrored version of an original video stream captured by a video capturing device.

[0043] In block 323, method 300 may include determining whether the video stream comprises video content showing a shape having the specified shape characteristics of the at least one shape. If there is a particular shape that matches at least one of the stored shapes based on comparing their characteristics, method 300 may proceed to block 324 where character data within the shape may be recognized and/or detected. On the other hand, if method 300 determines that there is no such shape found in the video stream, the method 300 may proceed to block 330. Method 300 may then stop in block 330.

[0044] Referring back to FIG. 1, detecting engine 122 may be responsible for implementing method 300.

[0045] FIG. 4 is a flow diagram depicting an example method 400 for video processing as used in a see-through screen-based collaboration system. Method 400 as illustrated (and described in greater detail below) is meant be an example and,

as such, should not be viewed as limiting. Method 400 may be implemented in the form of executable instructions stored on a machine-readable storage medium, such as storage medium 210, and/or in the form of electronic circuitry.

[0046] Method 400 may start in block 410 and proceed to block 421 where any content being shared between the local and remote users may be projected to a first see-through screen for a first user. The first user may modify, add, or otherwise interact with the shared content on the screen. A video image of the first user (and the space around the first user) may be received (block 422) and used to generate a mirrored video image (block 423). In block 424, method 400 may include recognizing character data in the mirrored video image. Various different character recognition techniques, as discussed herein, may be used. In block 425, a portion representing the character data may be flipped around an axis (e.g., the central axis of the portion) in the mirrored video image. In block 426, an output video image comprising the mirrored video image with the flipped portion may be generated. The output video image may be combined with the shared content (as modified, added, or otherwise interacted with by the first user) and projected to a second see-through screen for a second user (block 427). Method 400 may then stop in block 430.

[0047] Referring back to FIG. 1, output generating engine 125 may be responsible for implementing blocks 421, 426, and 427. Video mirroring engine 121 may be responsible for implementing blocks 422 and 423. Detecting engine 122 may be responsible for implementing block 424. Correcting engine 123 may be responsible for implementing block 425.

[0048] FIG. 5 is an example picture 500 depicting how two users communicate using a see-through screen-based collaboration system. Note the video image of the remote user is horizontally flipped so that drawings (e.g., shared content) made on the screen appear correct to both users. In this example, the words written on the remote user's T-shirt may be detected and flipped around according to various implementations as discussed herein.

[0049] FIG. 6 is a diagram depicting an example system design 600 of a see-through screen-based collaboration system.

[0050] The see-through screen-based collaboration system may include a first video capturing device 635A that captures a view of a first user 610A through a first see-through screen 620A. The first video capturing device 635A may be installed behind the first screen 620A, shooting through the first screen 620A. The first user 610A may be present on the other side of the screen 620A, facing the first camera 635A through the first screen 620A. A first projector 630A may be installed on the same side as the first camera 635A, projecting shared content 650 on the first screen 620A.

[0051] A similar arrangement of the system may be set up at a remote location for a second user 610B. For example, a second video capturing device 635B may capture a view of the second user 610B through a second see-through screen 620B. The second video capturing device 635B may be installed behind the second screen 620B, shooting through the second screen 620B. The second user 610B may be present on the other side of the screen 620B, facing the second camera 635B through the second screen 620B. The content being shared and interacted with by the first user 610A as well as the captured view of the first user 610A may be projected by a second projector 630B to the second see-through screen 620B. Likewise, the content being shared and interacted with by the second user 610B as well as the captured view of the second user 610B may be projected by the first projector 630A to the first see-through screen 620A. This particular arrangement of the see-through screen-based system would allow for capturing a video image of the remote user from a viewpoint that corresponds to that of the local user.

[0052] The first video capturing device 635A may capture a video image 640 of the first user 610A. As discussed above, because the shared content 650 will be displayed in the same orientation for both users, system 110 may create a mirror version 641 of the video image 640 so that the shared content displayed on the screen appear correct to both users. One problem with this is that any characters

shown in the mirrored video image 641 will appear to be written backwards. To correct this undesirable mirror image effect, system 110 may recognize and/or detect character data (e.g., letters, numbers, symbols, etc.) in the mirrored video image 641 and flip a portion of the mirrored video image 641 representing the detected character data (e.g., a corrected video image 642). In FIG. 6, the word "HELLO" written on the first user 610A's T-shirt may be detected and flipped around according to various implementations as discussed herein. Further, resolution and accuracy of the word "HELLO" may be enhanced according to various implementations as discussed herein.

[0053] System 110 may then generate an output video stream 643 to be projected to the second see-through screen 620B. The output video stream 643 may comprise the corrected video image 642 and the shared content 650. The shared content 650 may be combined with the corrected video stream 642 to create the output video stream 643. The output video stream 643 may then be projected to and/or displayed on the second see-through screen 620B. Not shown in FIG. 6 is the reverse path where the second user 610B's video image is captured by the second camera 635B, mirrored, corrected, and displayed with the shared content 650 on the first see-through screen 620A.

[0054] FIG. 7 is a diagram depicting an example implementation of detecting character data in a video stream.

[0055] A view captured by a video capturing device may have a designated section 760 (shown shaded in FIG. 7) of the view where any character data within the designated section 760 may be detected. A user 710 can hold a sheet of paper 750 up so that any character data (e.g., the word "HELLO") written on the paper 750 is positioned within the designated section 760 of the view. System 110 may recognize and/or detect the word "HELLO" shown within the designated section 760 of the camera view. System 110 may create a mirrored version 741 of the captured video 740 and further correct the mirrored version 741 by horizontally flipping the word "HELLO" (e.g., a corrected video image 742).

[0056] FIG. 8 is a diagram depicting an example implementation 800 of controlling a mirror image effect using a depth camera.

[0057] A depth camera 830 may determine the distance to a human 810 or other object in a field of view of the camera 830. Using the depth camera 830, system 110 may mirror reverse (or flip horizontally) any characters, physical objects, or even the entire space that are between a first distance from the depth camera 830 to a second distance from the depth camera 830. In some instances, the viewing space of the camera may be divided into several different depth ranges. For each depth range, system 110 may specify what needs to be mirror reversed and what needs to remain unchanged (e.g., not mirrored). For example, system 110 may mirror reverse (or flip horizontally) only the character data that is present within a first depth range 850. The orientation of any other physical objects or the space itself within that depth range may remain unchanged. Further, the entire space within a second depth range 851 may be mirror reversed (or horizontally flipped) whereas the orientation of the space within a third depth range 852 may remain the same (e.g., not mirrored).

[0058] The present invention has been shown and described with reference to the foregoing examples. It is to be understood, however, that other forms, details and examples may be made without departing from the spirit and scope of the invention that is defined in the following claims.

CLAIMS

1. A method for execution by a computing device for detecting character data, the method comprising:

storing at least one shape and characteristics related to the at least one shape in a database;

obtaining a video stream;

determining whether the video stream comprises video content showing a shape having the characteristics of the at least one shape; and

in response to determining that the video stream comprises video content showing a shape having the characteristics of the at least one shape, detecting the character data within the shape.

2. The method of claim 1, wherein the video stream is a mirrored version of an original video stream captured by a video capturing device, the method further comprising:

flipping a portion of the video stream representing the detected character data around a first axis; and

generating an output video stream comprising the video stream having the flipped portion.

3. The method of claim 1, wherein the at least one shape comprise a square, rectangle, octagon, T-shirt shape, or a flag shape.

4. The method of claim 2, further comprising:

correcting perspective distortion caused by the flipping of the portion representing the detected character data.

5. The method of claim 1, further comprising:

enhancing the detected character data by enhancing resolution or accuracy of the detected character data; and

generating an output video stream comprising the video stream having the enhanced character data.

6. A machine-readable storage medium comprising instructions executable by a processor of a computing device for recognizing character data, the machine-readable storage medium comprising:

instructions for receiving a video image from a first camera that captures the video image of a view through a first see-through screen;

instructions for generating a mirrored video image using the captured video image;

instructions for recognizing character data in the mirrored video image; and

instructions for flipping around a first axis a portion representing the recognized character data in the mirrored video image.

7. The machine-readable storage medium of claim 6, wherein the instructions for recognizing the character data in the mirrored video image further comprise instructions for recognizing the character data based on OCR (optical character recognition) technology.

8. The machine-readable storage medium of claim 6, wherein the instructions for recognizing the character data in the mirrored video image further comprise:

instructions for storing at least one shape and characteristics related to the at least one shape in a database;

instructions for determining whether the mirrored video image comprises video content showing a shape having the characteristics of the at least one shape; and

in response to determining that the mirrored video image comprises video content showing a shape having the characteristics of the at least one shape, instructions for recognizing the character data within the shape.

9. The machine-readable storage medium of claim 6, wherein the instructions for recognizing the character data in the mirrored video image further comprise:

instructions for determining when any character data is presented within a designated section of the view captured by the first camera; and

instructions for recognizing the character data within the designated section.

10. The machine-readable storage medium of claim 6, wherein the first camera is a depth camera and the view includes a plurality of objects, the method further comprising:

instructions for detecting a presence of at least one object between a first distance from the depth camera and a second distance from the depth camera; and

instructions for generating the mirrored video image of the at least one object.

11. The machine-readable storage medium of claim 6, further comprising:

instructions for providing shared content to be projected to the first see-through screen; and

instructions for generating an output video image that is to be projected to a second see-through screen, wherein the output video image comprises the mirrored video image with the flipped portion and the shared content.

12. A system for detecting character data comprising:

a processor configured to:

obtain a real-time video stream from a video capturing device that captures a view of a first user through a first see-through screen, wherein

the first user and a second user communicate using see-through screen-based collaboration systems;

mirror video content of the real-time video stream to create a mirrored version of the real-time video stream;

detect the character data in the mirrored version;

flip a portion representing the detected character data horizontally in the mirrored version;

generate an output video stream that includes the mirrored version with the flipped portion; and

provide the output video stream to the second user such that the output video stream is projected to a second see-through screen used by the second user.

13. The system of claim 12, wherein detecting the character data in the mirrored version comprises:

storing at least one shape and characteristics related to the at least one shape in a database;

determining whether the mirrored version comprises video content showing a shape having the characteristics of the at least one shape; and

in response to determining that the mirrored version comprises video content showing a shape having the characteristics of the at least one shape, detecting the character data within the shape.

14. The system of claim 12, wherein detecting the character data in the mirrored version comprises:

determining when any character data is presented within a designated section of the view captured by the video capturing device; and

detecting the character data within the designated section.

15. The system of claim 12, wherein the processor is further configured to:

enhance the detected character data by enhancing resolution or accuracy of the detected character data; and

generate the output video stream that includes the enhanced character data.

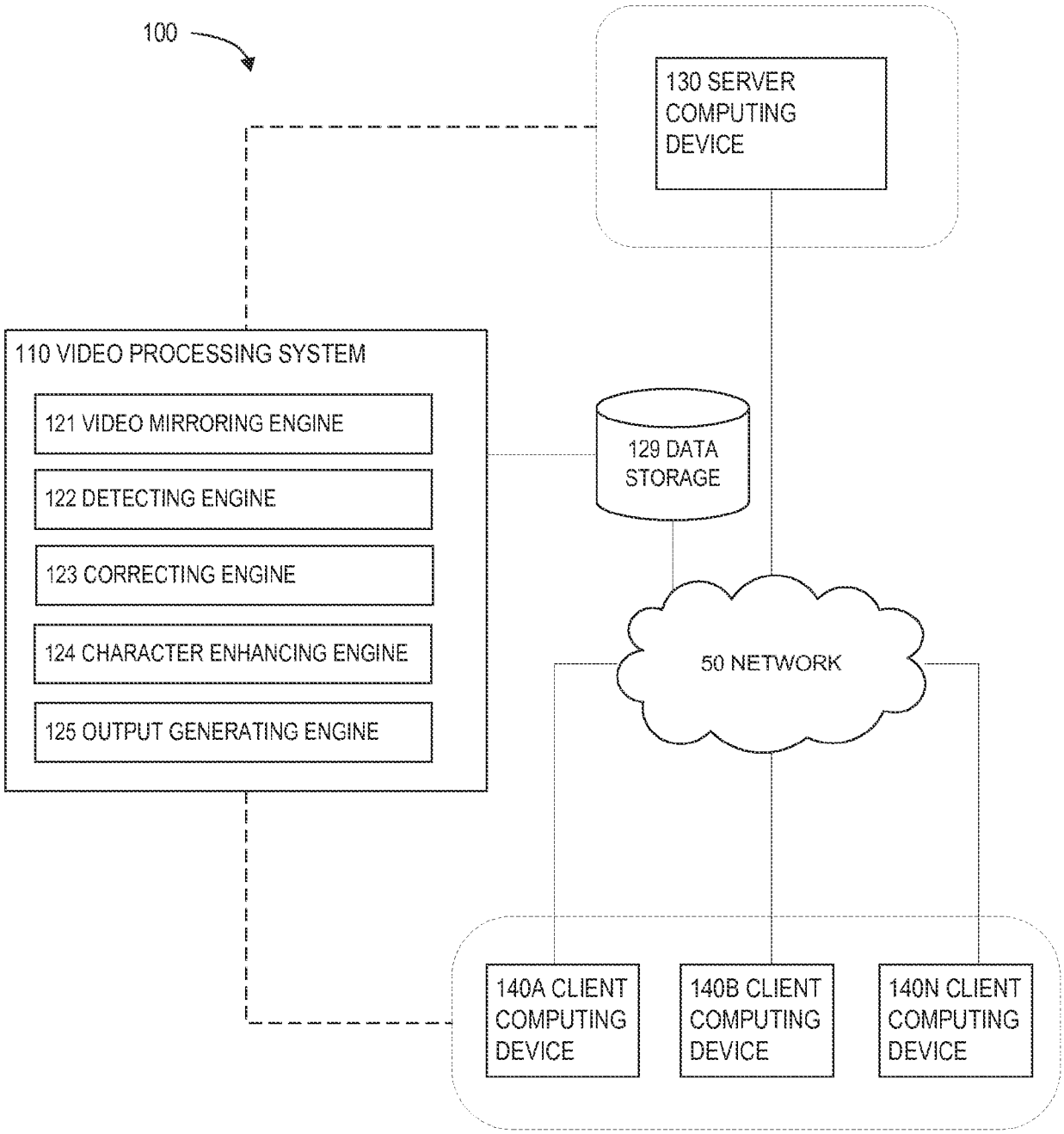

100
130 SERVER COMPUTING DEVICE
110 VIDEO PROCESSING SYSTEM
121 VIDEO MIRRORING ENGINE
122 DETECTING ENGINE
123 CORRECTING ENGINE
124 CHARACTER ENHANCING ENGINE
125 OUTPUT GENERATING ENGINE
129 DATA STORAGE
50 NETWORK
140A CLIENT COMPUTING DEVICE
140B CLIENT COMPUTING DEVICE
140N CLIENT COMPUTING DEVICE

FIG. 1

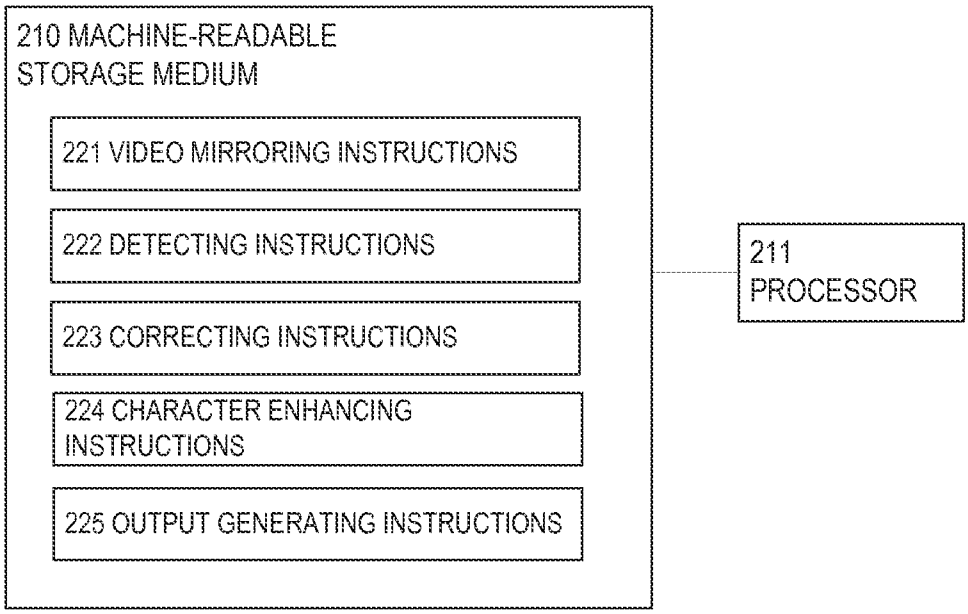

210 MACHINE-READABLE STORAGE MEDIUM
221 VIDEO MIRRORING INSTRUCTIONS
222 DETECTING INSTRUCTIONS
223 CORRECTING INSTRUCTIONS
224 CHARACTER ENHANCING INSTRUCTIONS
225 OUTPUT GENERATING INSTRUCTIONS
211 PROCESSOR

FIG. 2

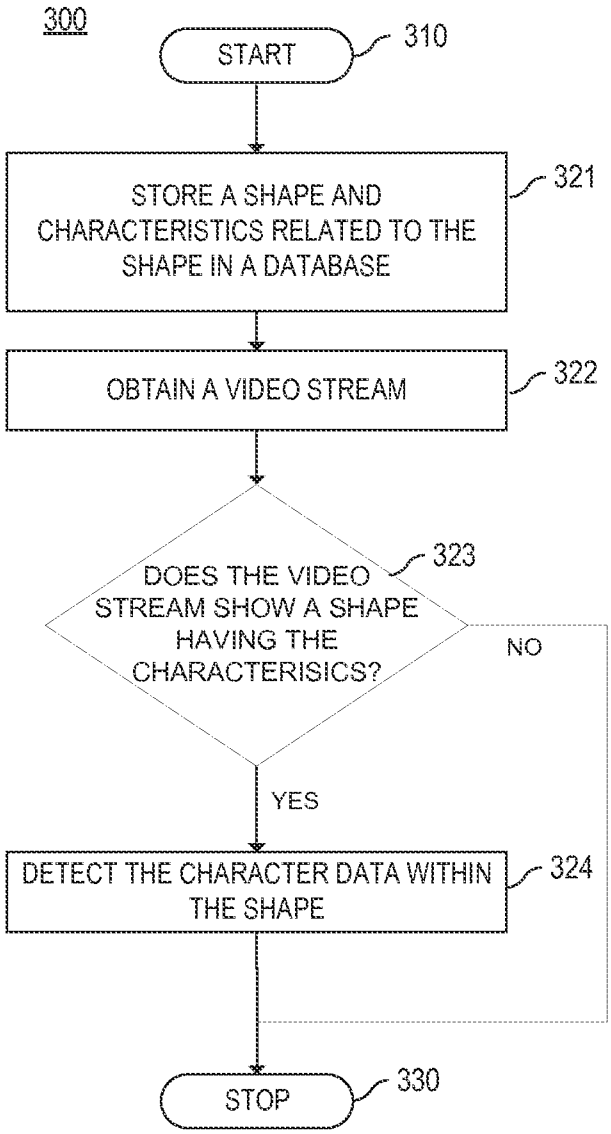

300
START
310
STORE A SHAPE AND CHARACTERISTICS RELATED TO THE SHAPE IN A DATABASE
321
OBTAIN A VIDEO STREAM
322
DOES THE VIDEO STREAM SHOW A SHAPE HAVING THE CHARACTERISICS?
323
NO
YES
DETECT THE CHARACTER DATA WITHIN THE SHAPE
324
STOP
330

FIG. 3

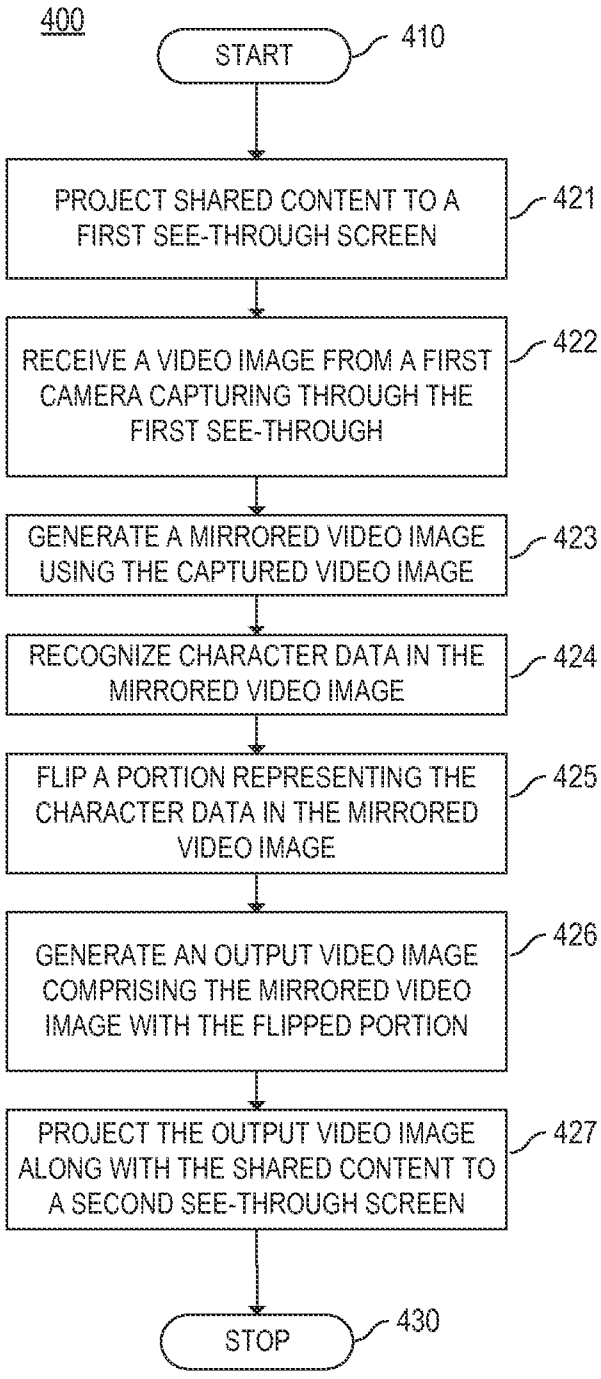

400
START
410
PROJECT SHARED CONTENT TO A FIRST SEE-THROUGH SCREEN
421
RECEIVE A VIDEO IMAGE FROM A FIRST CAMERA CAPTURING THROUGH THE FIRST SEE-THROUGH
422
GENERATE A MIRRORED VIDEO IMAGE USING THE CAPTURED VIDEO IMAGE
423
RECOGNIZE CHARACTER DATA IN THE MIRRORED VIDEO IMAGE
424
FLIP A PORTION REPRESENTING THE CHARACTER DATA IN THE MIRRORED VIDEO IMAGE
425
GENERATE AN OUTPUT VIDEO IMAGE COMPRISING THE MIRRORED VIDEO IMAGE WITH THE FLIPPED PORTION
426
PROJECT THE OUTPUT VIDEO IMAGE ALONG WITH THE SHARED CONTENT TO A SECOND SEE-THROUGH SCREEN
427
STOP
430

FIG. 4

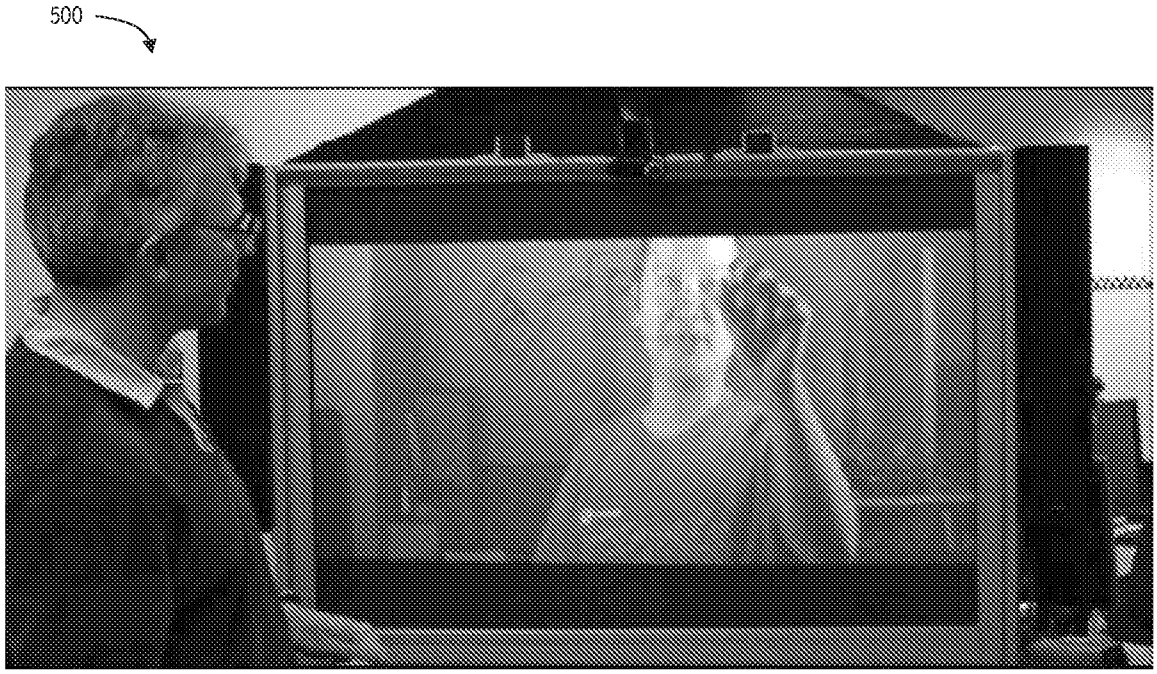


FIG. 5

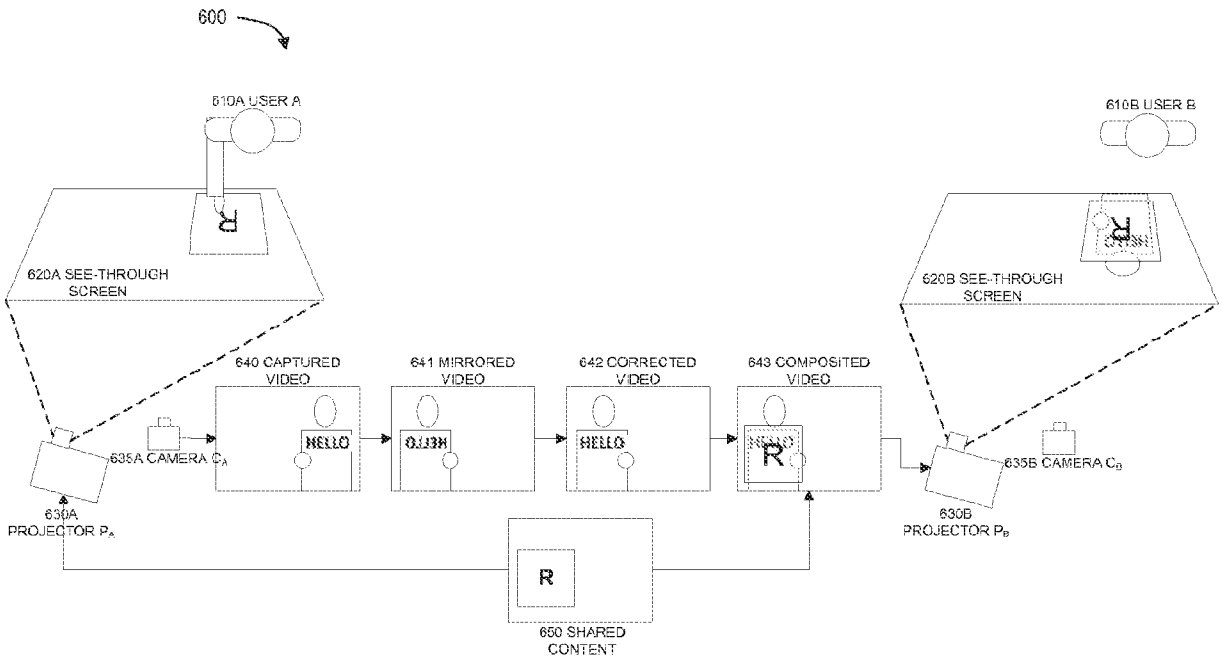

600
610A USER A
610B USER B
620A SEE-THROUGH SCREEN
620B SEE-THROUGH SCREEN
640 CAPTURED VIDEO
641 MIRRORED VIDEO
642 CORRECTED VIDEO
643 COMPOSITED VIDEO
635A CAMERA C_A
635B CAMERA C_B
630A PROJECTOR P_A
630B PROJECTOR P_B
650 SHARED CONTENT
HELLO
R

FIG. 6

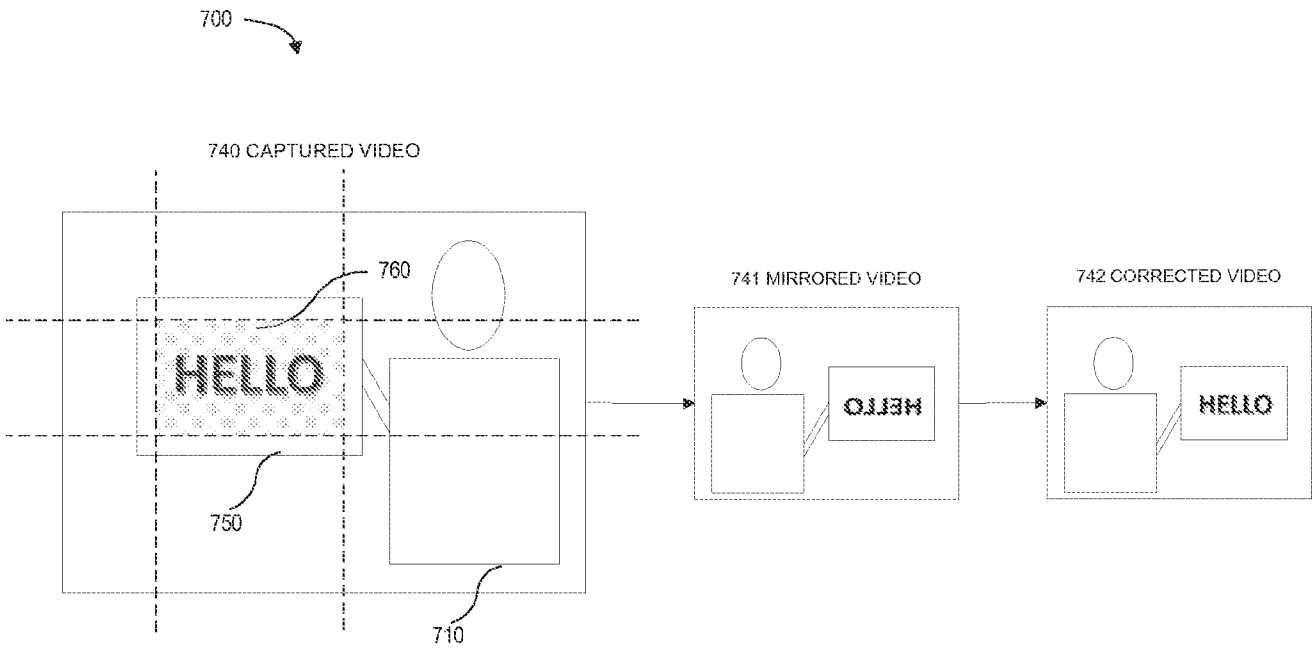

700
740 CAPTURED VIDEO
760
HELLO
750
710
741 MIRRORED VIDEO
HELLO
742 CORRECTED VIDEO
HELLO

FIG. 7

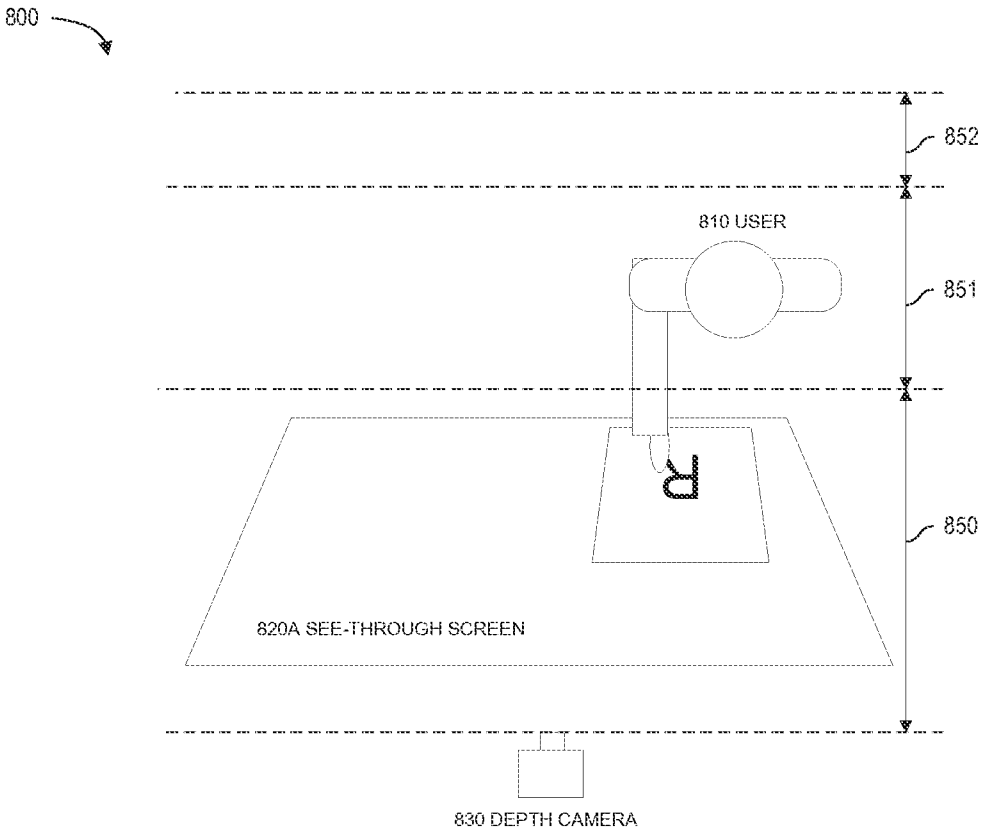

800
852
810 USER
851
R
850
820A SEE-THROUGH SCREEN
830 DEPTH CAMERA

FIG. 8

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2014/044969

A. CLASSIFICATION OF SUBJECT MATTER

H04N 7/15(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04N 7/15; G06K 9/36; H04N 9/74; H04N 7/18; G09G 5/00; G06F 3/01

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & keywords: mirror, image, flip, text

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2008-0292215 A1 (DALE ELLEN GAUCAS et al.) 27 November 2008 See paragraphs [0023], [0034]-[0068] and figures 1, 4A, 5, 7.	1-15
Y	US 2013-0278629 A1 (KAR-HAN TAN) 24 October 2013 See paragraphs [0013]-[0021] and figures 1, 7.	1-15
A	US 2012-0284646 A1 (DAVID H. SITRICK et al.) 08 November 2012 See paragraphs [0323]-[0329], claims 1-2 and figures 30-31, 33-36.	1-15
A	US 2004-0080616 A1 (MAMORU MIYAWAKI et al.) 29 April 2004 See abstract, paragraphs [0061]-[0086] and figures 1-2.	1-15
A	US 2005-0162511 A1 (WARREN BRUCE JACKSON) 28 July 2005 See abstract, claims 1-20 and figures 5-6.	1-15

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents,such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
15 February 2015 (15.02.2015)	**24 February 2015 (24.02.2015)**

Name and mailing address of the ISA/KR
International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon Metropolitan City, 302-701,
Republic of Korea
Facsimile No. ++82 42 472 3473

Authorized officer
KIM, Seong Woo
Telephone No. +82-42-481-3348

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.
PCT/US2014/044969

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2008-0292215 A1	27/11/2008	None	
US 2013-0278629 A1	24/10/2013	None	
US 2012-0284646 A1	08/11/2012	US 08914735 B2	16/12/2014
US 2004-0080616 A1	29/04/2004	EP 0673161 A2	20/09/1995
		EP 0673161 A3	03/04/1996
		EP 0673161 B1	29/08/2001
		JP 3119413 B2	18/12/2000
		JP 3347461 B2	20/11/2002
		JP 7253571 A	03/10/1995
		JP 7311387 A	28/11/1995
		US 2001-0040549 A1	15/11/2001
		US 2008-0018809 A1	24/01/2008
		US 6232947 B1	15/05/2001
		US 6747613 B2	08/06/2004
		US 7304629 B2	04/12/2007
		US 7382335 B2	03/06/2008
US 2005-0162511 A1	28/07/2005	CN 1649419 A	03/08/2005
		DE 102004063577 A1	25/08/2005
		JP 2005-218103 A	11/08/2005
		JP 4257298 B2	22/04/2009
		US 7705877 B2	27/04/2010