



US 20110161843A1

(19) **United States**

(12) **Patent Application Publication**
Bennett et al.

(10) **Pub. No.: US 2011/0161843 A1**

(43) **Pub. Date: Jun. 30, 2011**

(54) **INTERNET BROWSER AND ASSOCIATED
CONTENT DEFINITION SUPPORTING
MIXED TWO AND THREE DIMENSIONAL
DISPLAYS**

Publication Classification

(51) **Int. Cl.**
G06F 3/00 (2006.01)

(52) **U.S. Cl.** **715/760**

(75) **Inventors:** **James D. Bennett**, Hroznetin (CZ);
Jeyhan Karaoguz, Irvine, CA (US)

(73) **Assignee:** **BROADCOM CORPORATION**,
Irvine, CA (US)

(21) **Appl. No.:** **12/982,140**

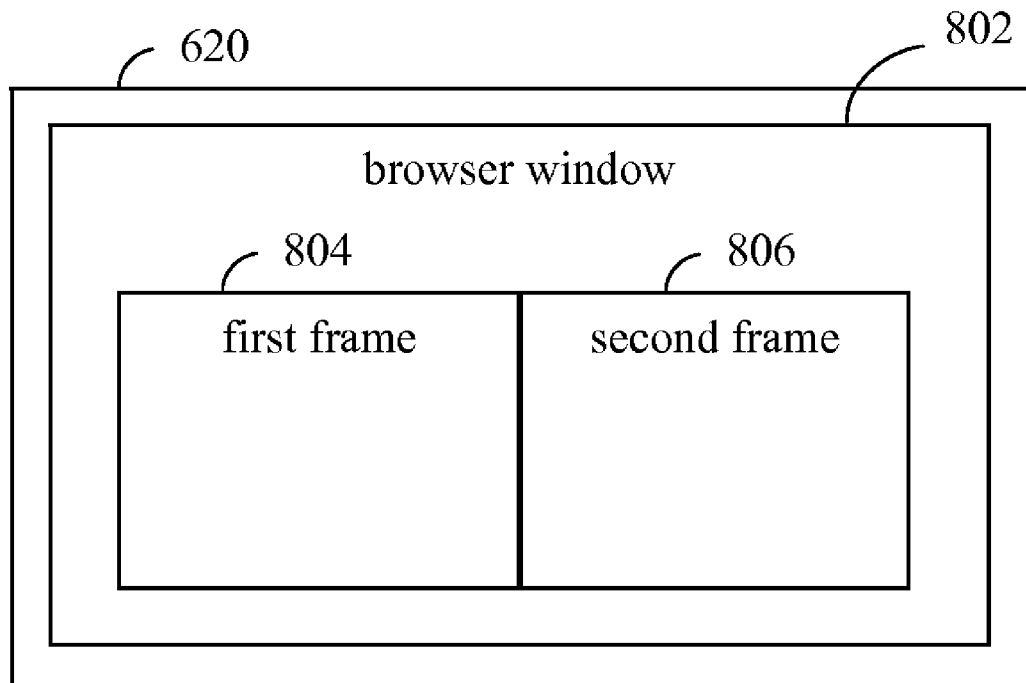
(22) **Filed:** **Dec. 30, 2010**

Related U.S. Application Data

(60) Provisional application No. 61/291,818, filed on Dec.
31, 2009, provisional application No. 61/303,119,
filed on Feb. 10, 2010.

(57) **ABSTRACT**

A browser architecture and associated content definition are provided that support display on a display screen of two-dimensional content and three-dimensional content. Web page content is received and parsed. Two-dimensional content to be displayed in a first region of the screen is identified. A first configuration request is communicated to cause a first configuration of the first region of the screen to support the two-dimensional content. Three-dimensional content to be displayed in a second region of the screen is identified. A second configuration request is communicated to cause a second configuration of the second region of the screen to support the three-dimensional content.



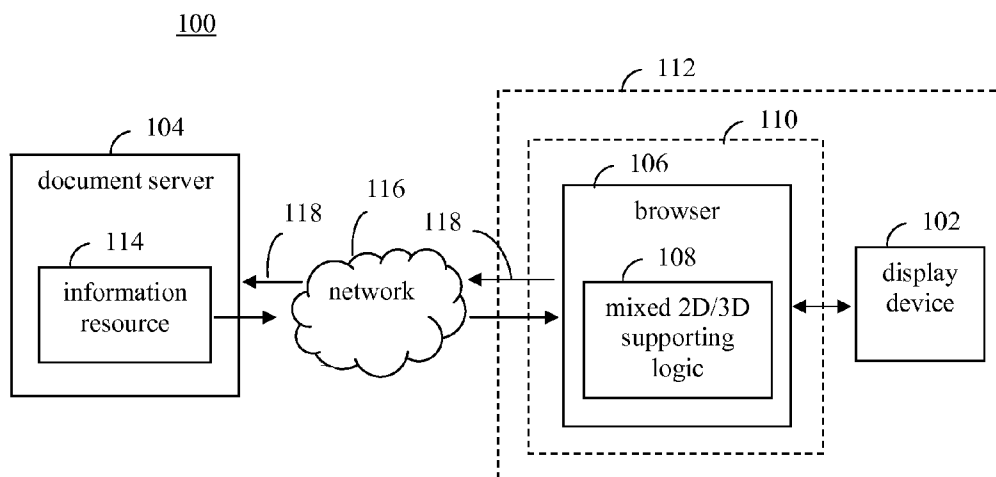


FIG. 1

200

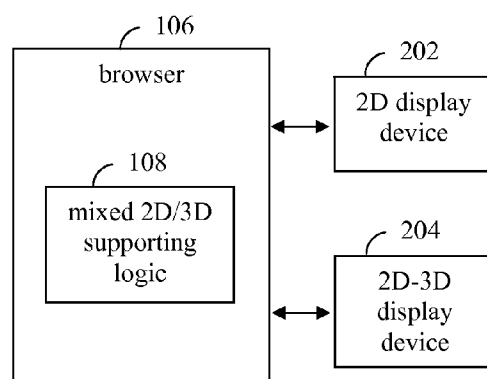


FIG. 2

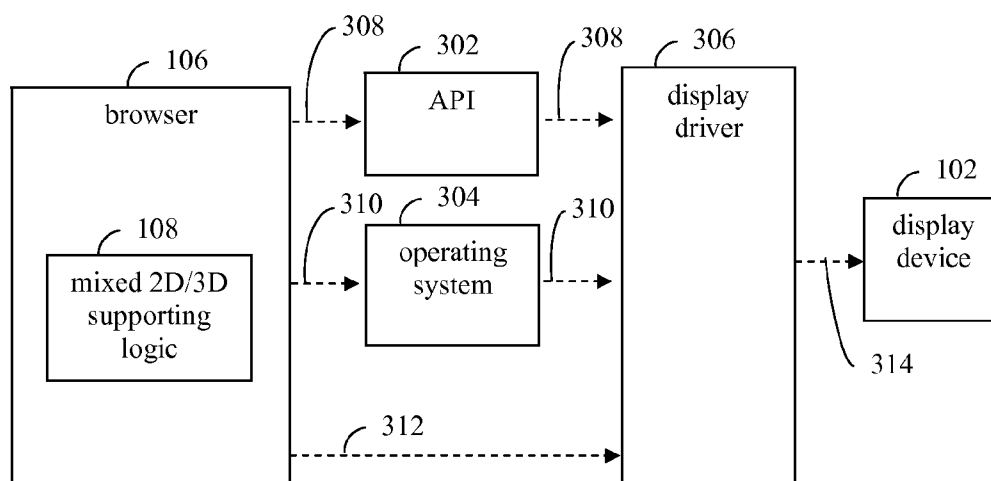


FIG. 3

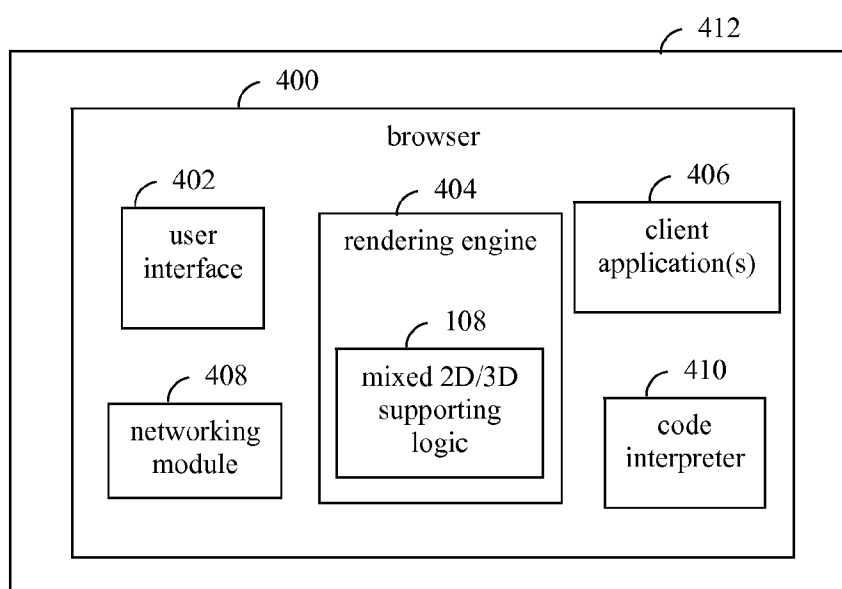


FIG. 4A

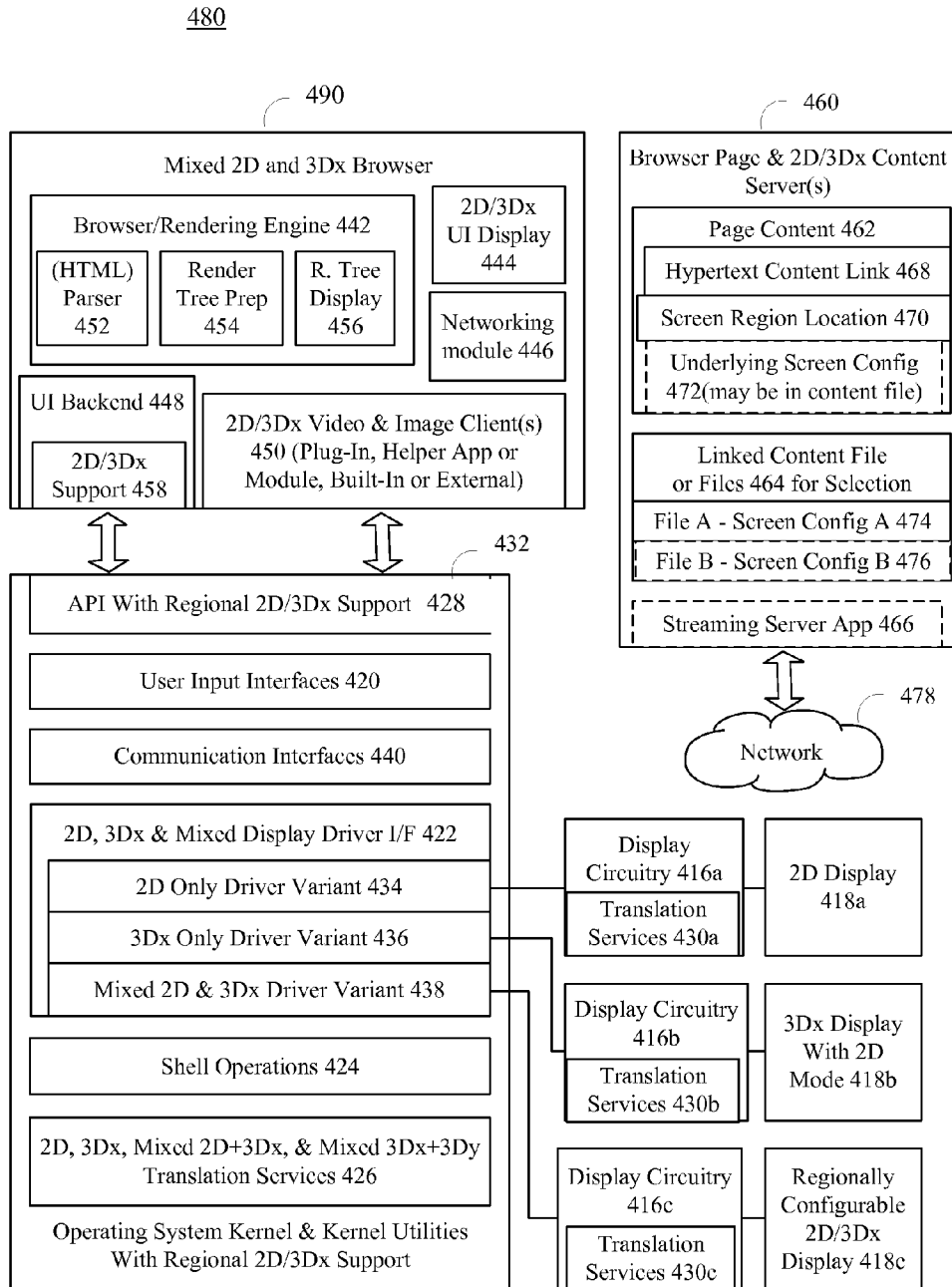


FIG. 4B

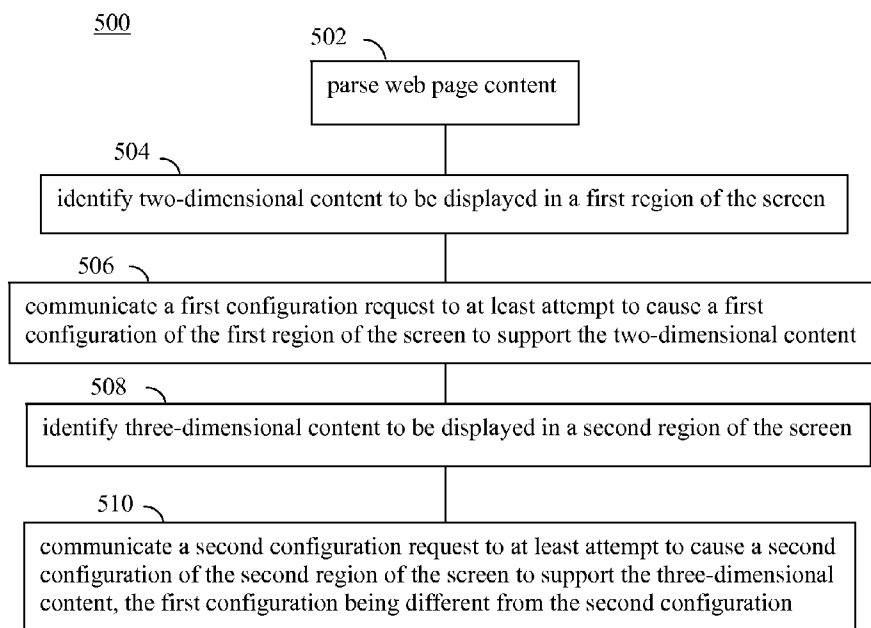


FIG. 5

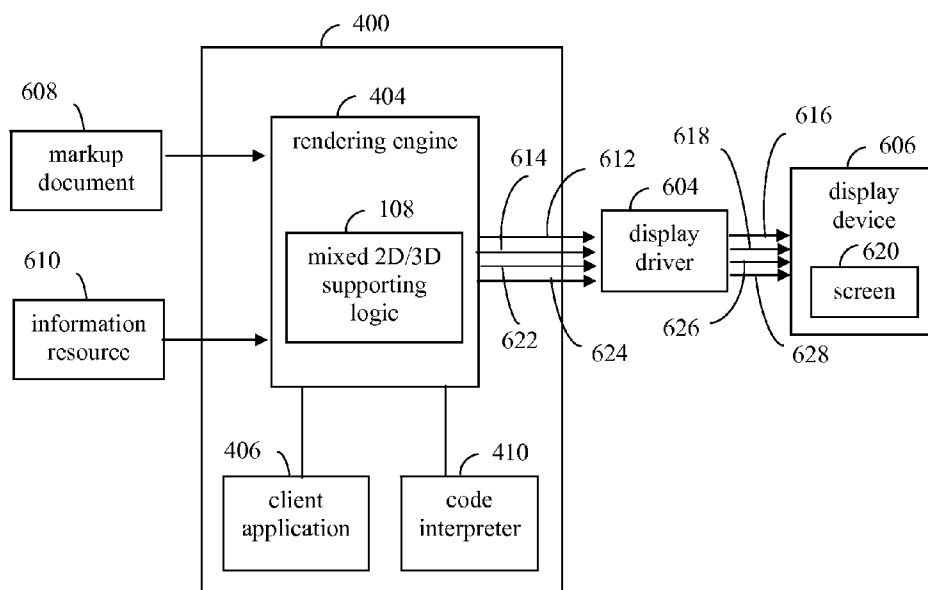


FIG. 6

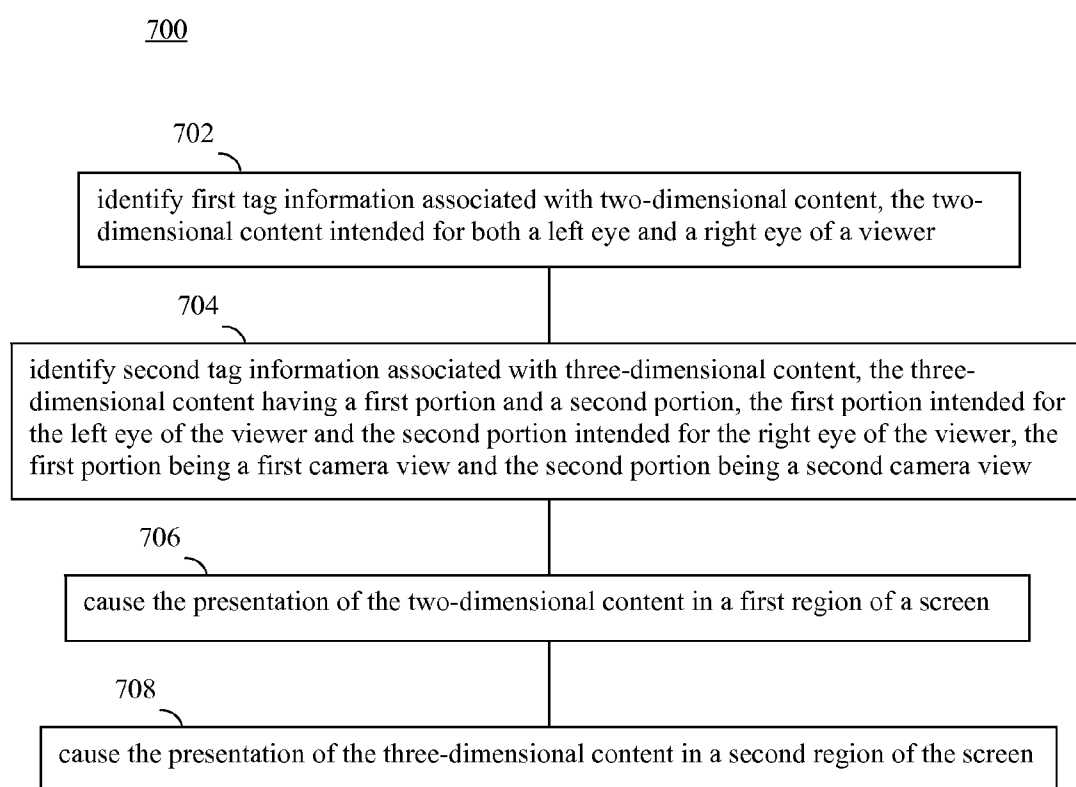


FIG. 7

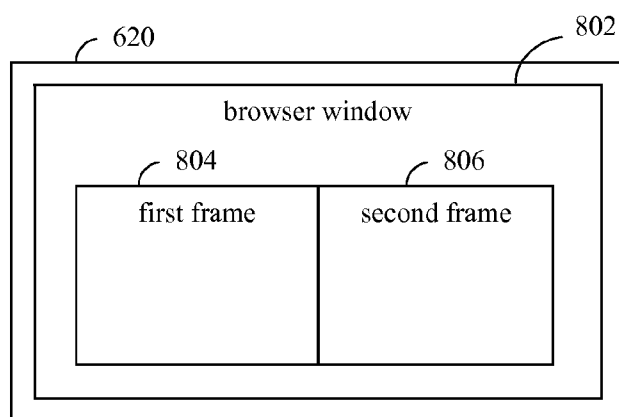


FIG. 8

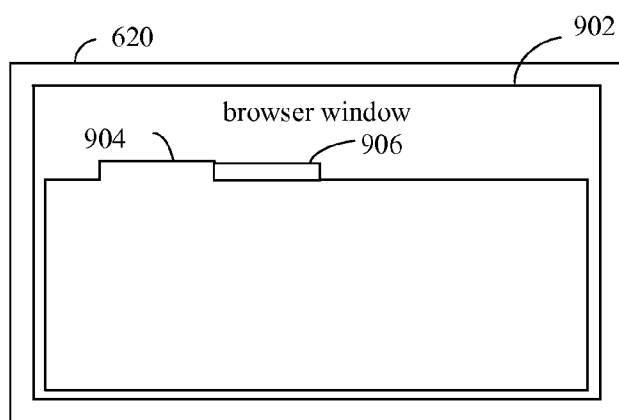


FIG. 9

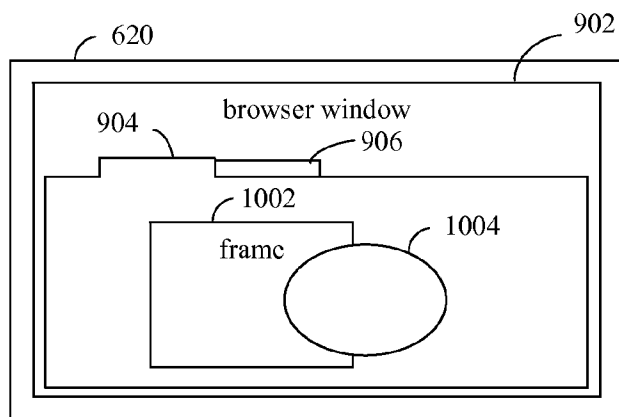


FIG. 10A

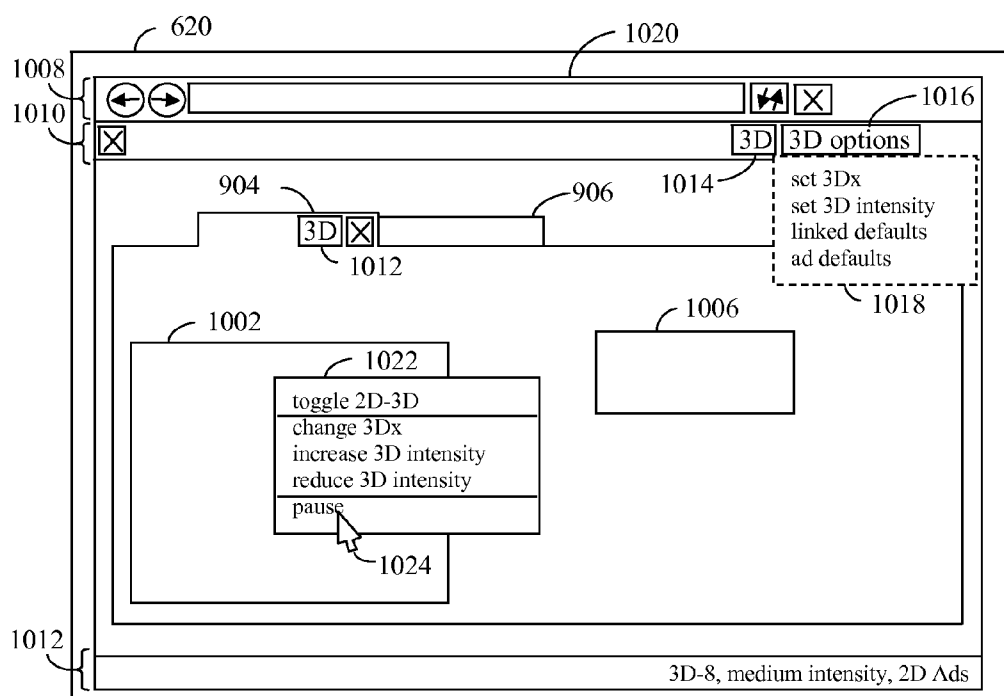


FIG. 10B

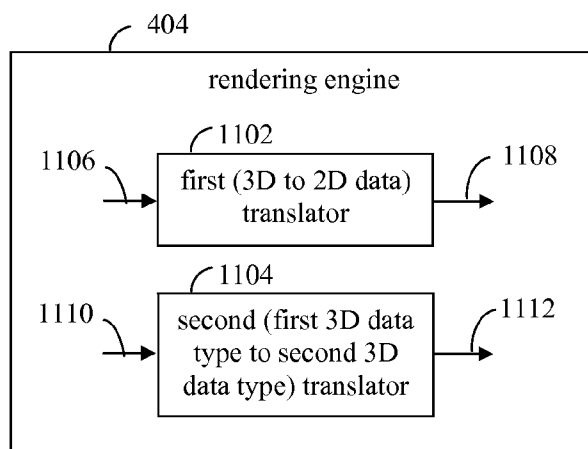


FIG. 11

1200

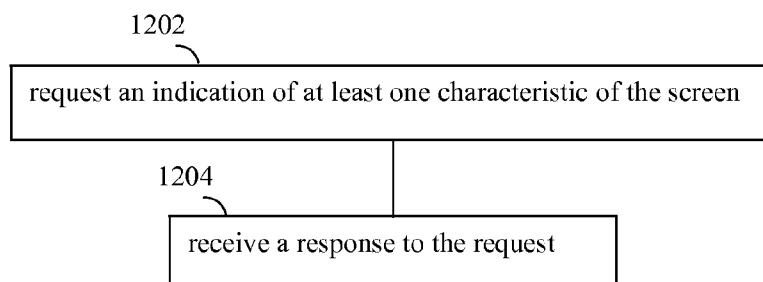


FIG. 12

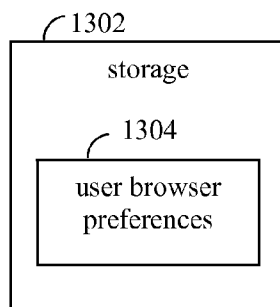


FIG. 13

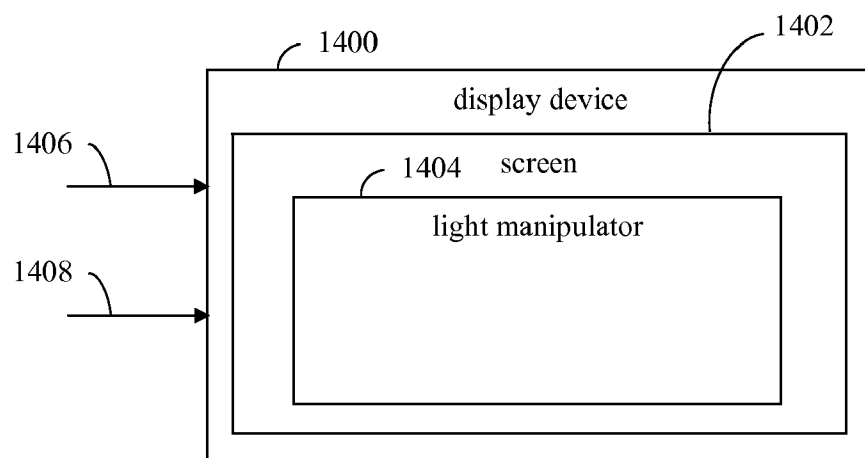


FIG. 14

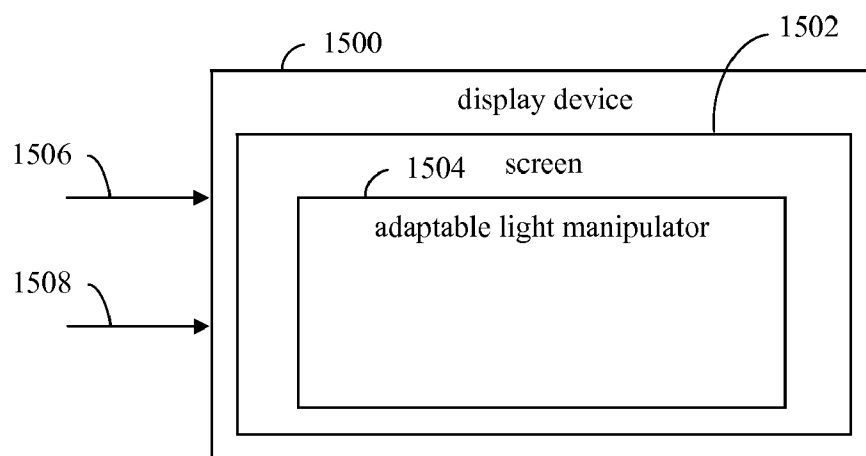


FIG. 15

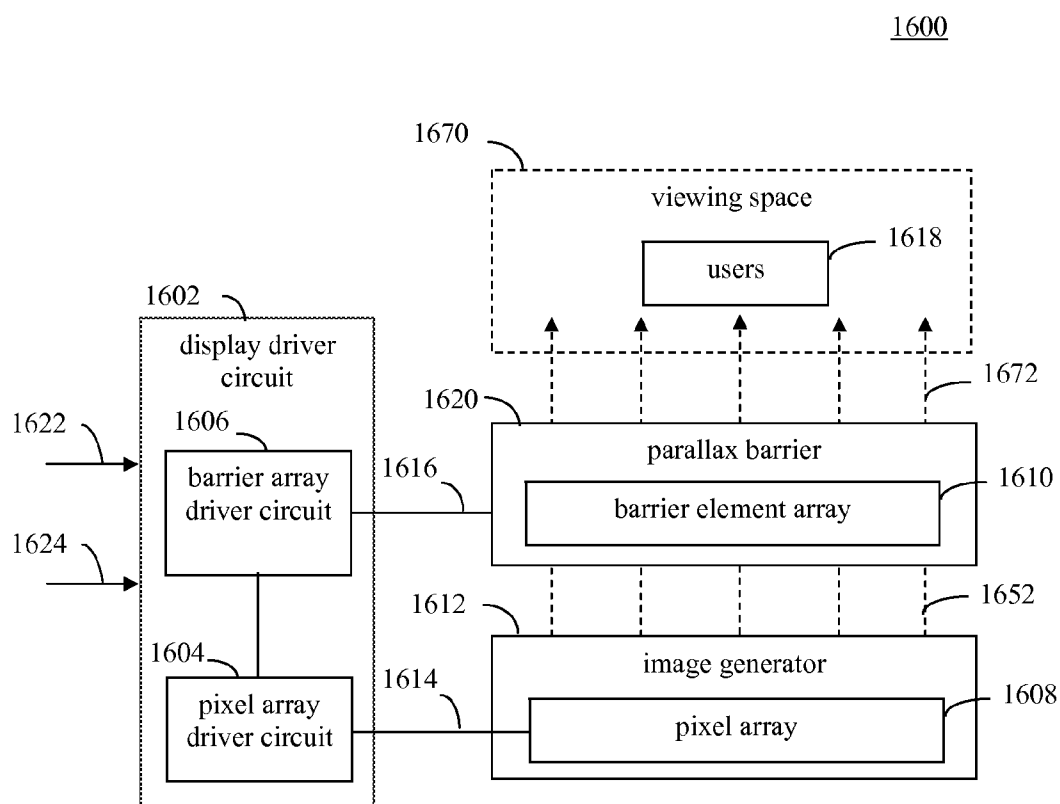


FIG. 16

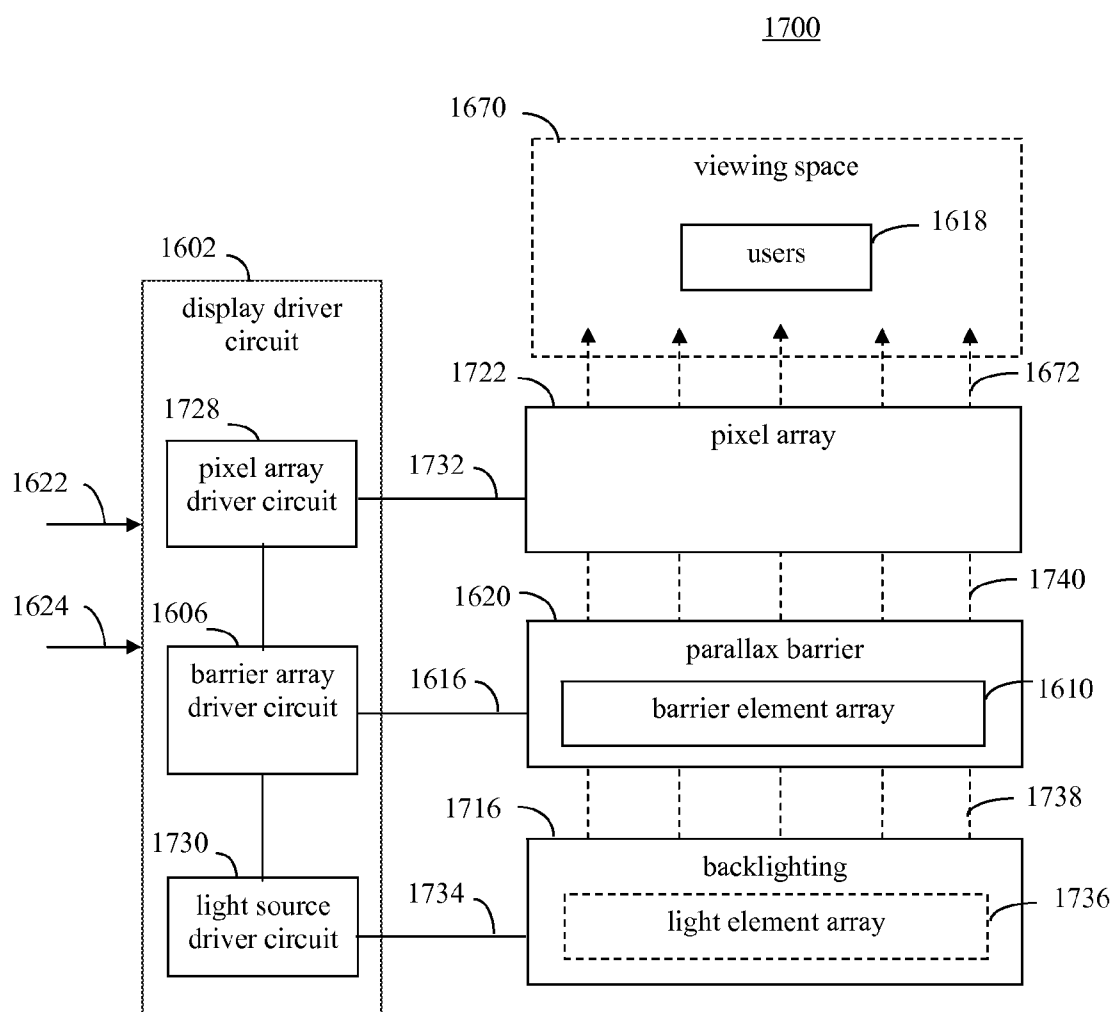


FIG. 17

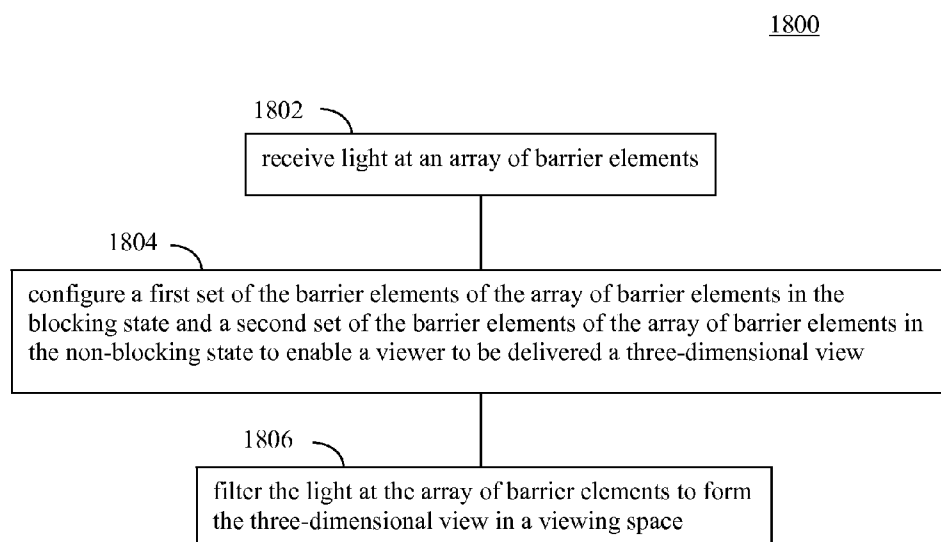


FIG. 18

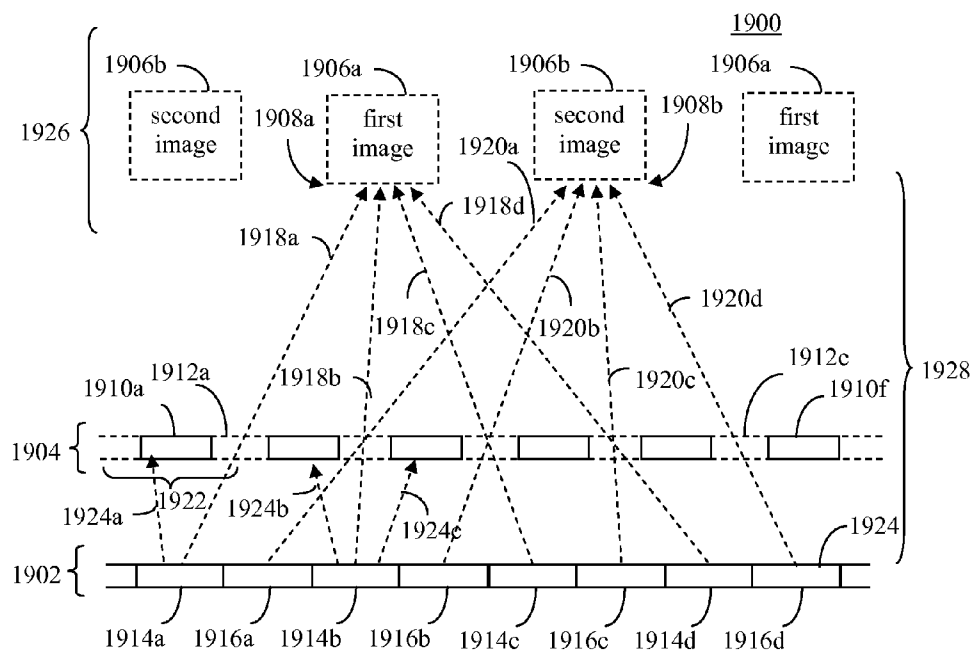


FIG. 19

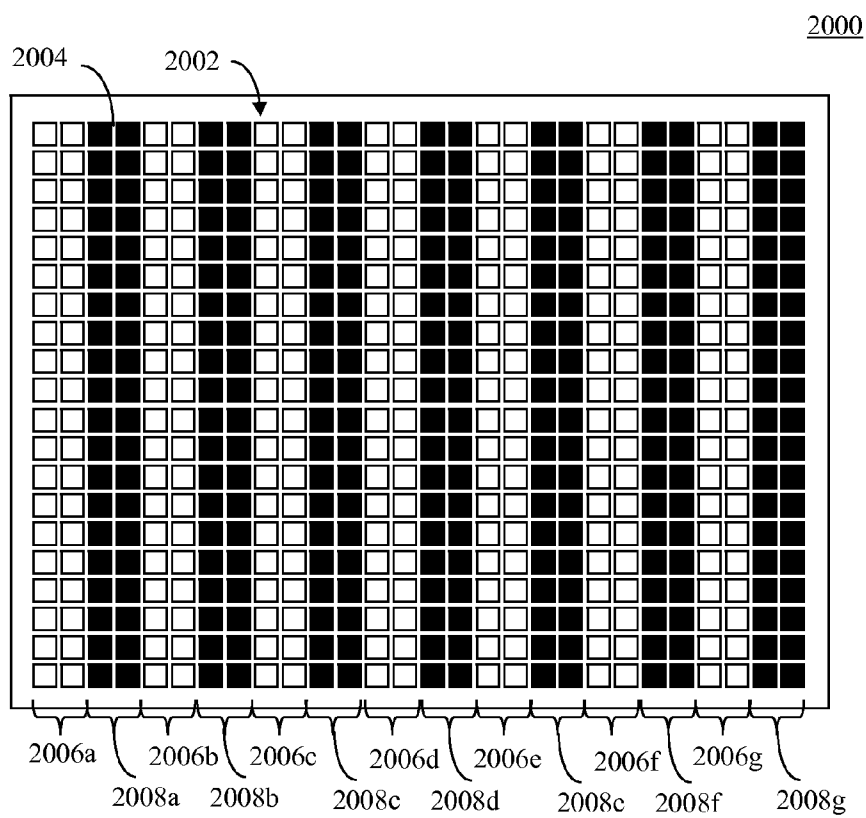


FIG. 20

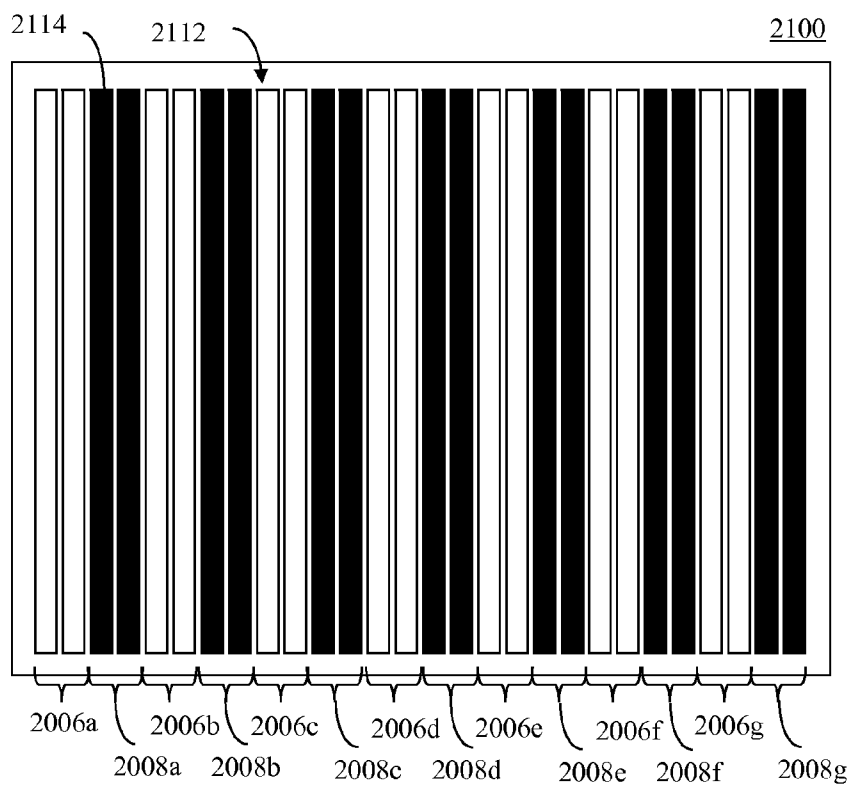


FIG. 21

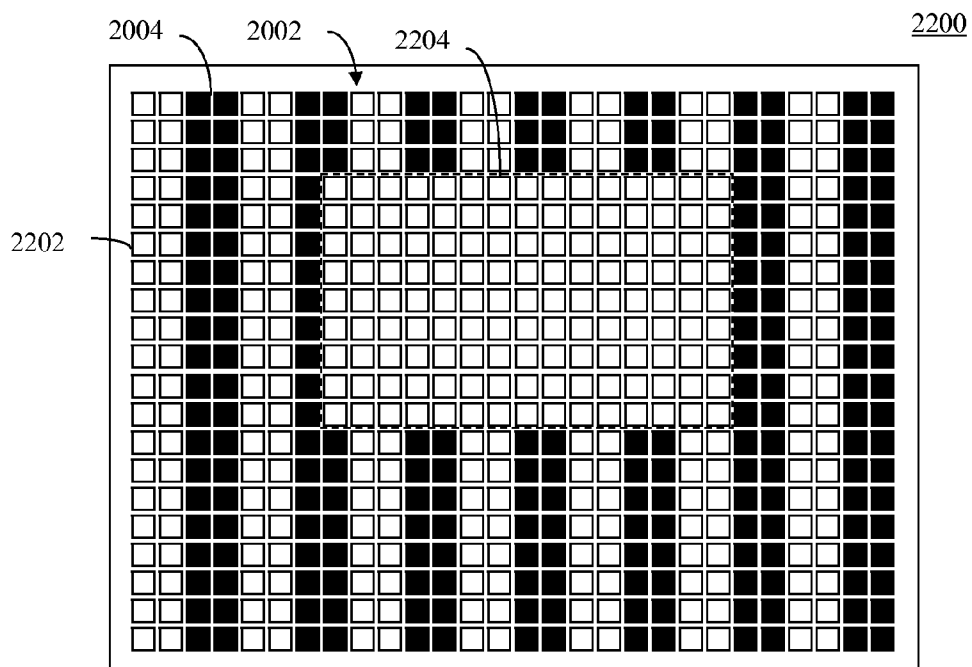


FIG. 22

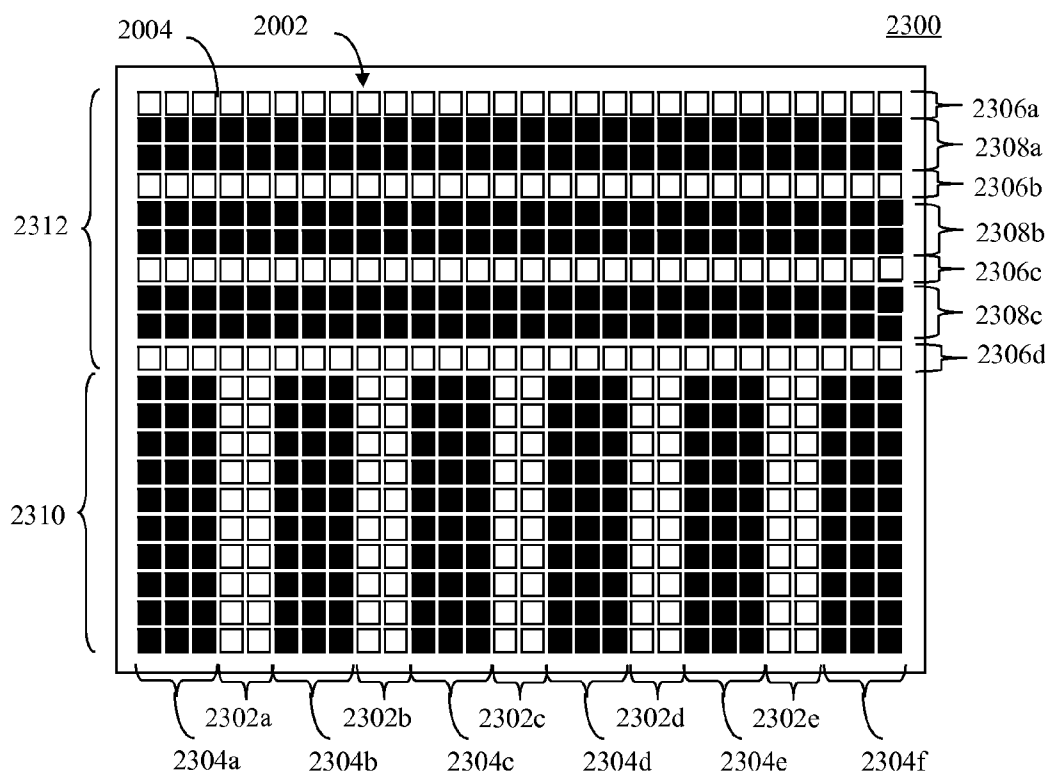


FIG. 23

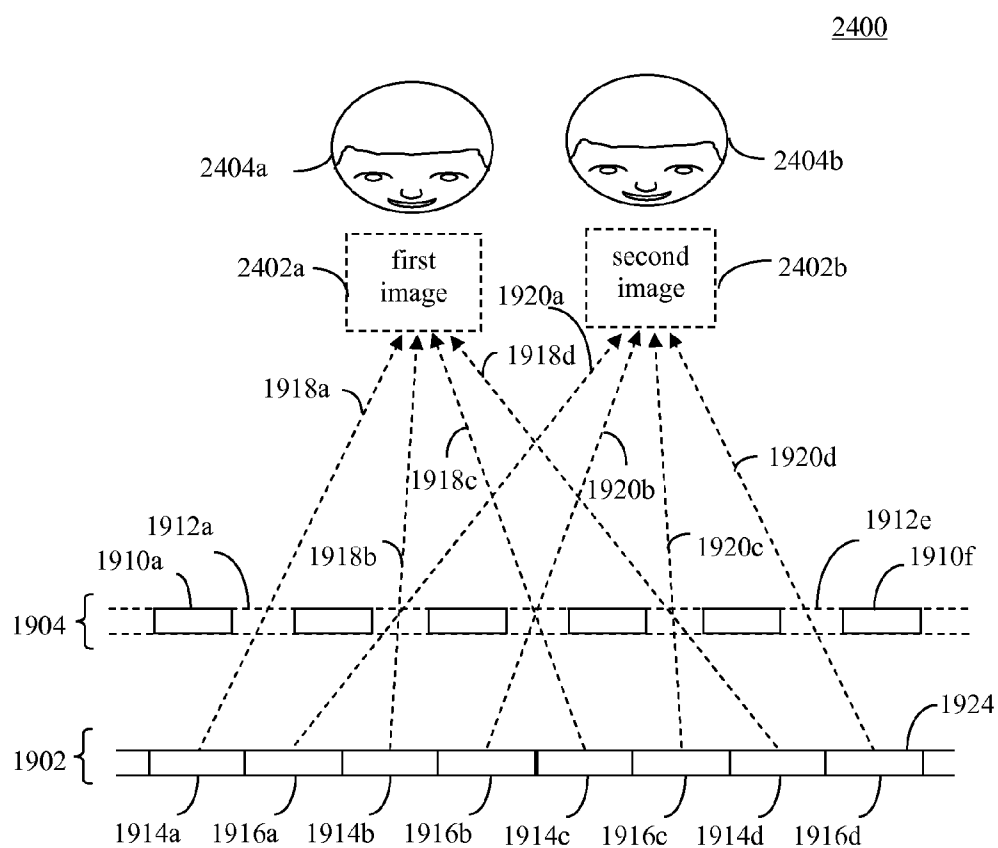


FIG. 24

FIG. 26

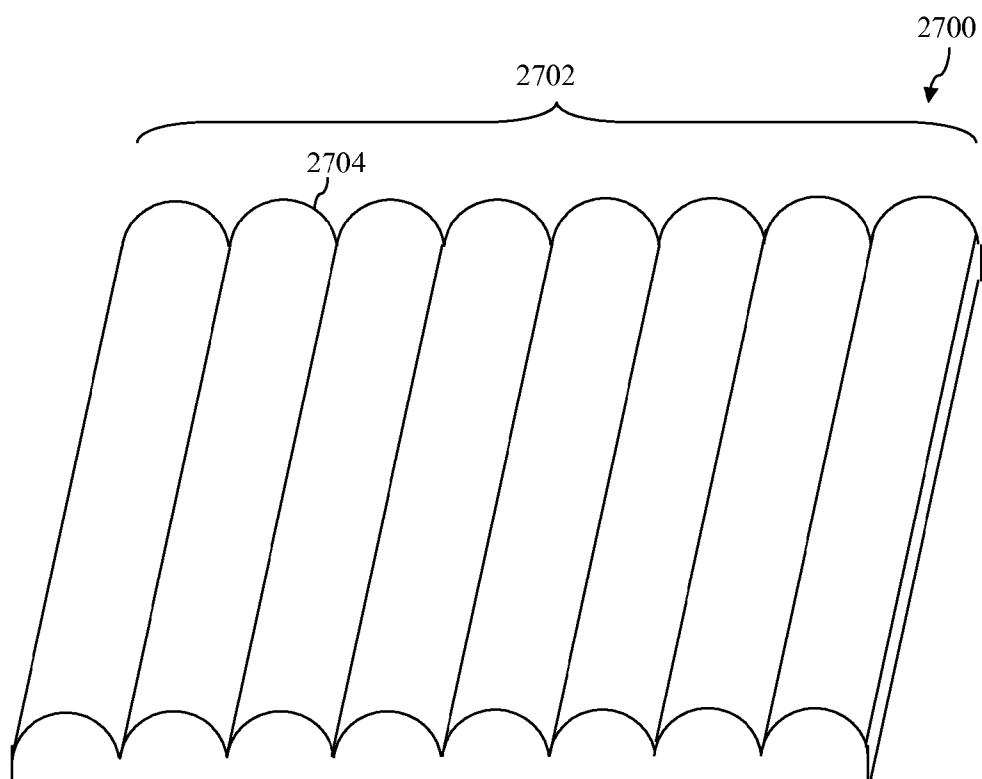


FIG. 27

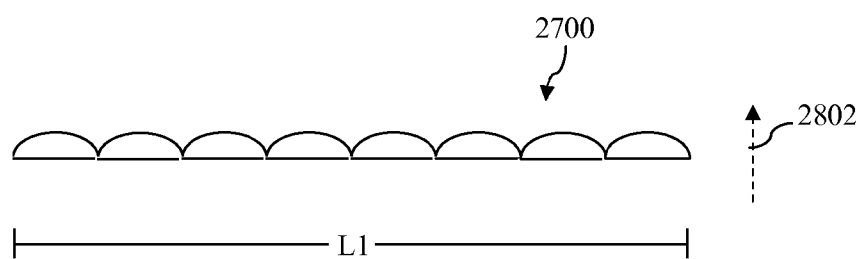


FIG. 28

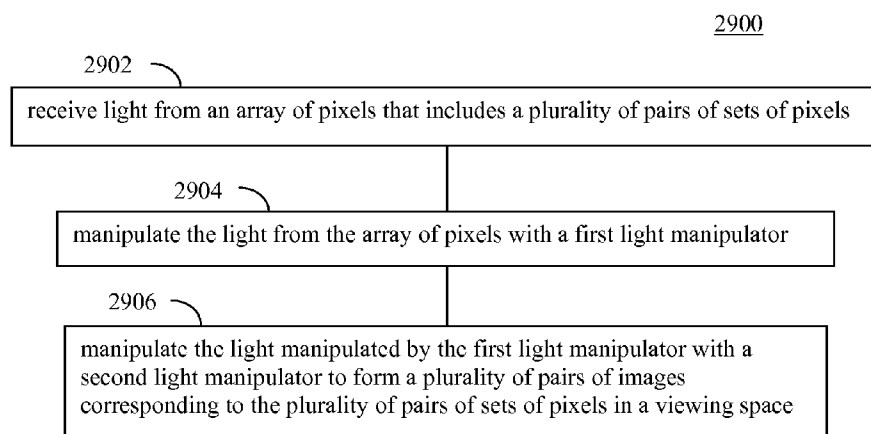


FIG. 29

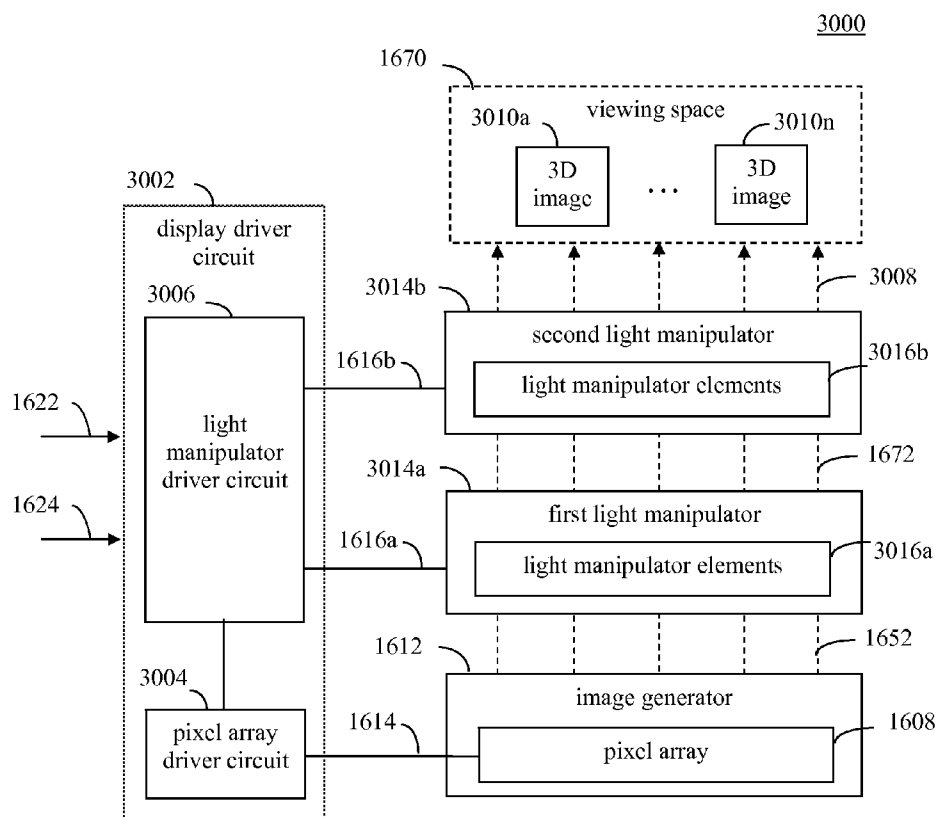


FIG. 30

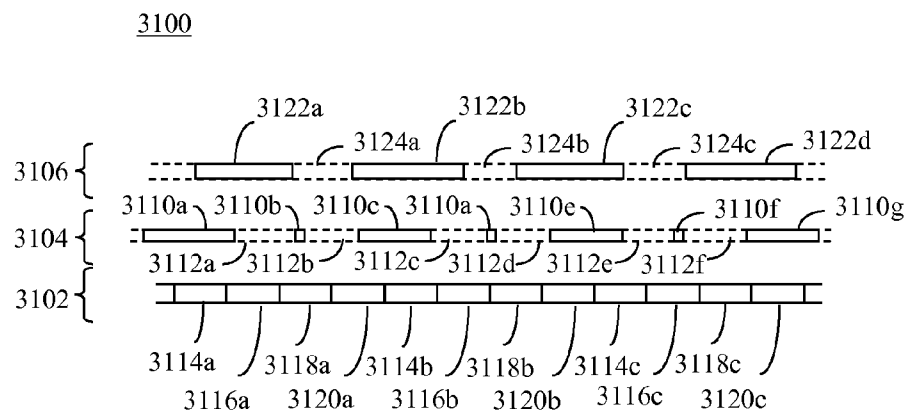


FIG. 31

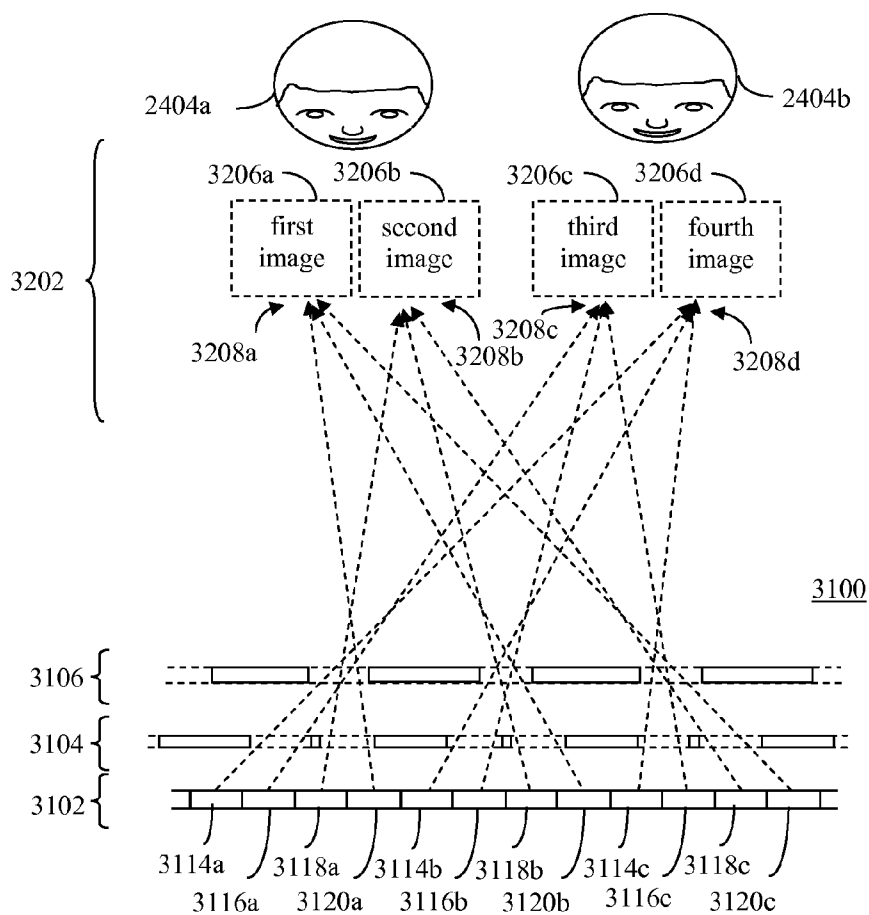


FIG. 32

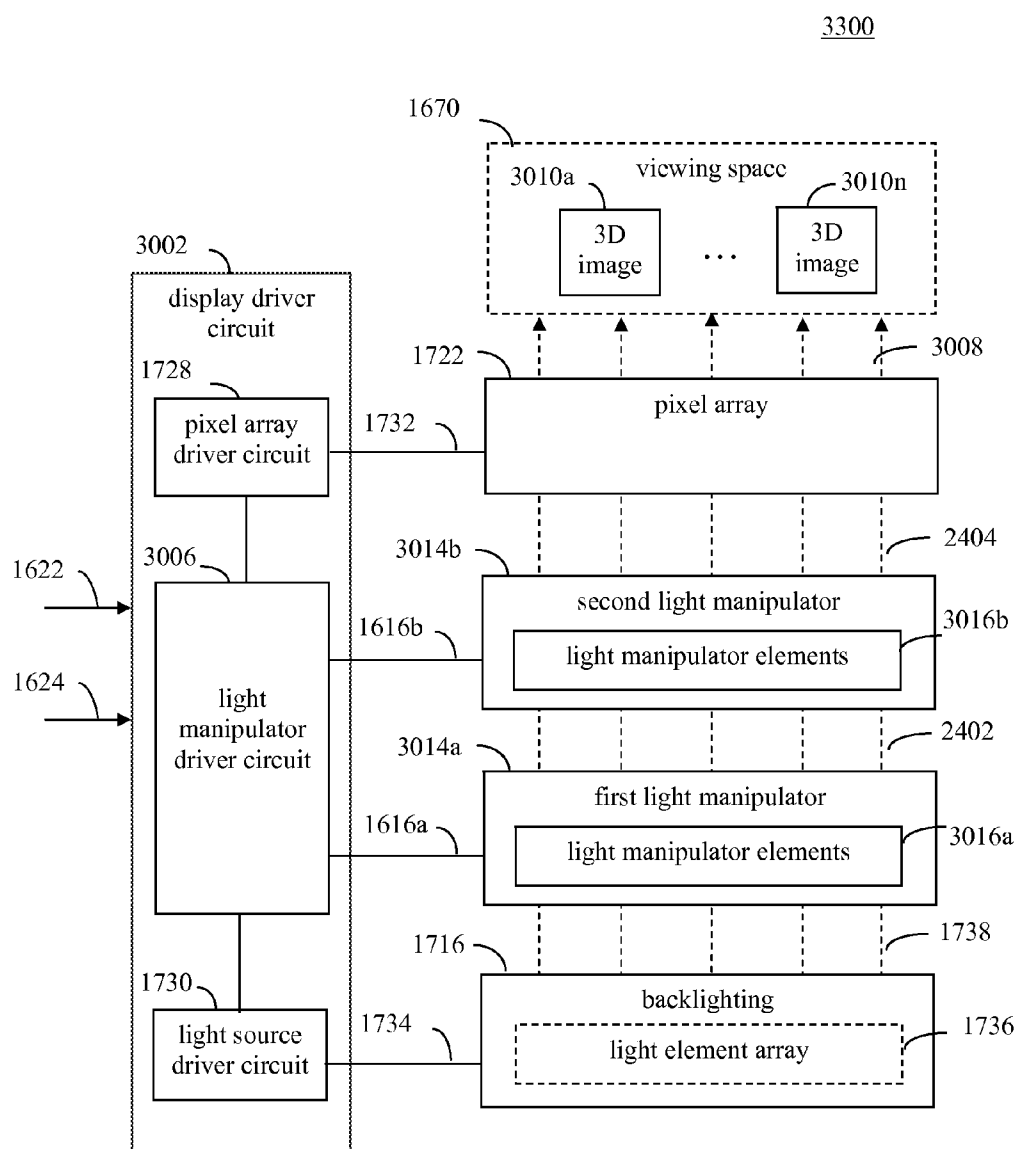


FIG. 33

3400

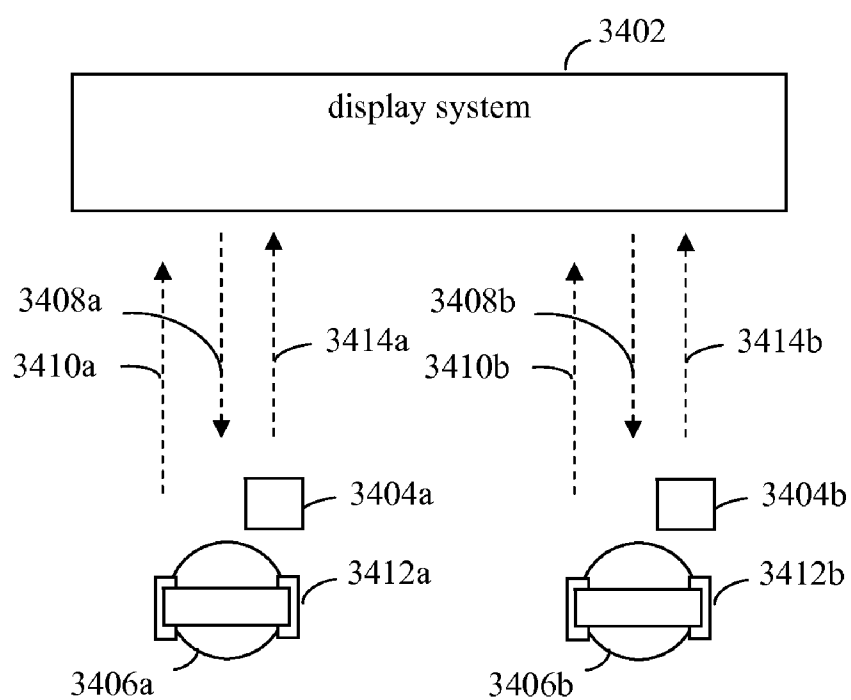


FIG. 34

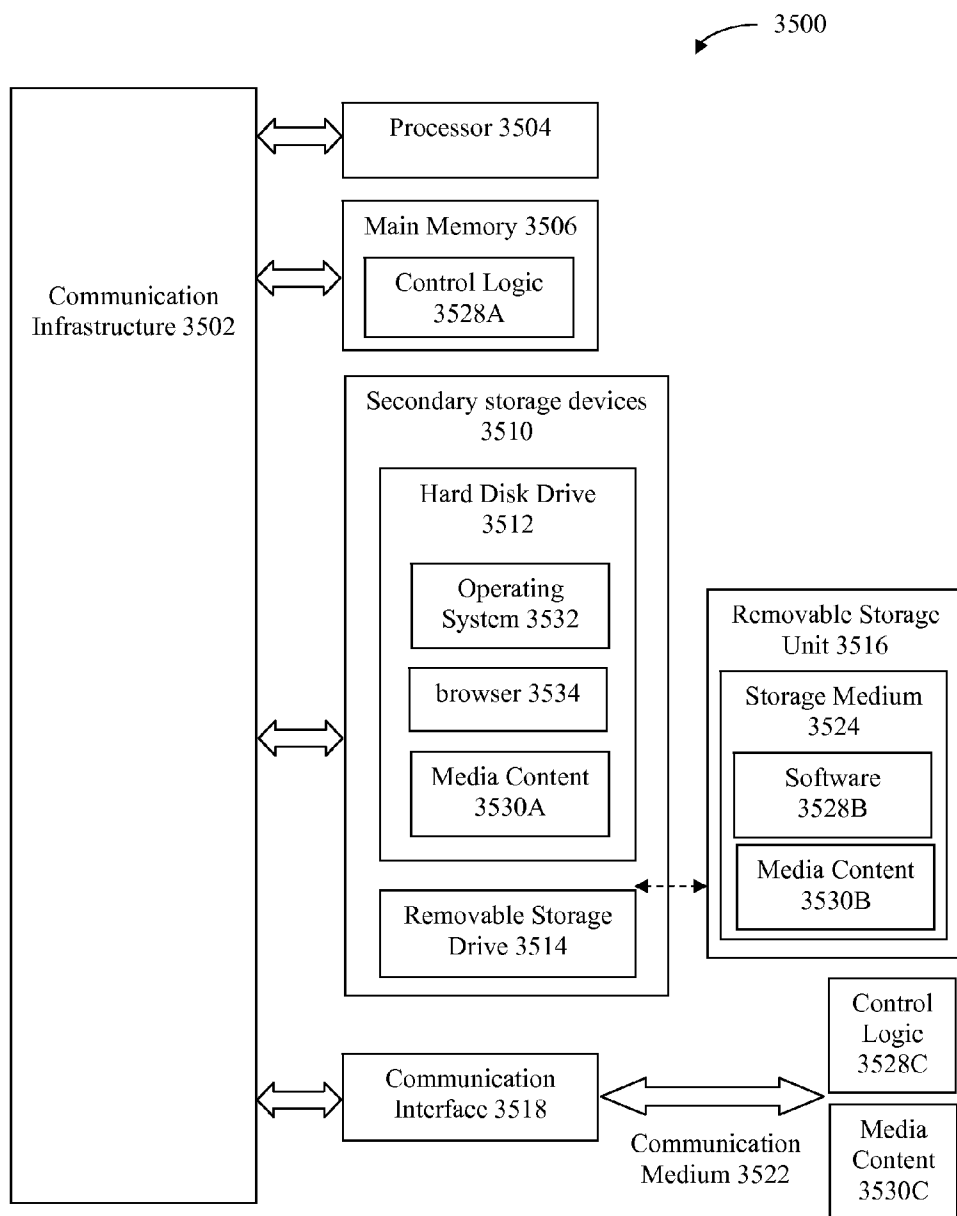


FIG. 35

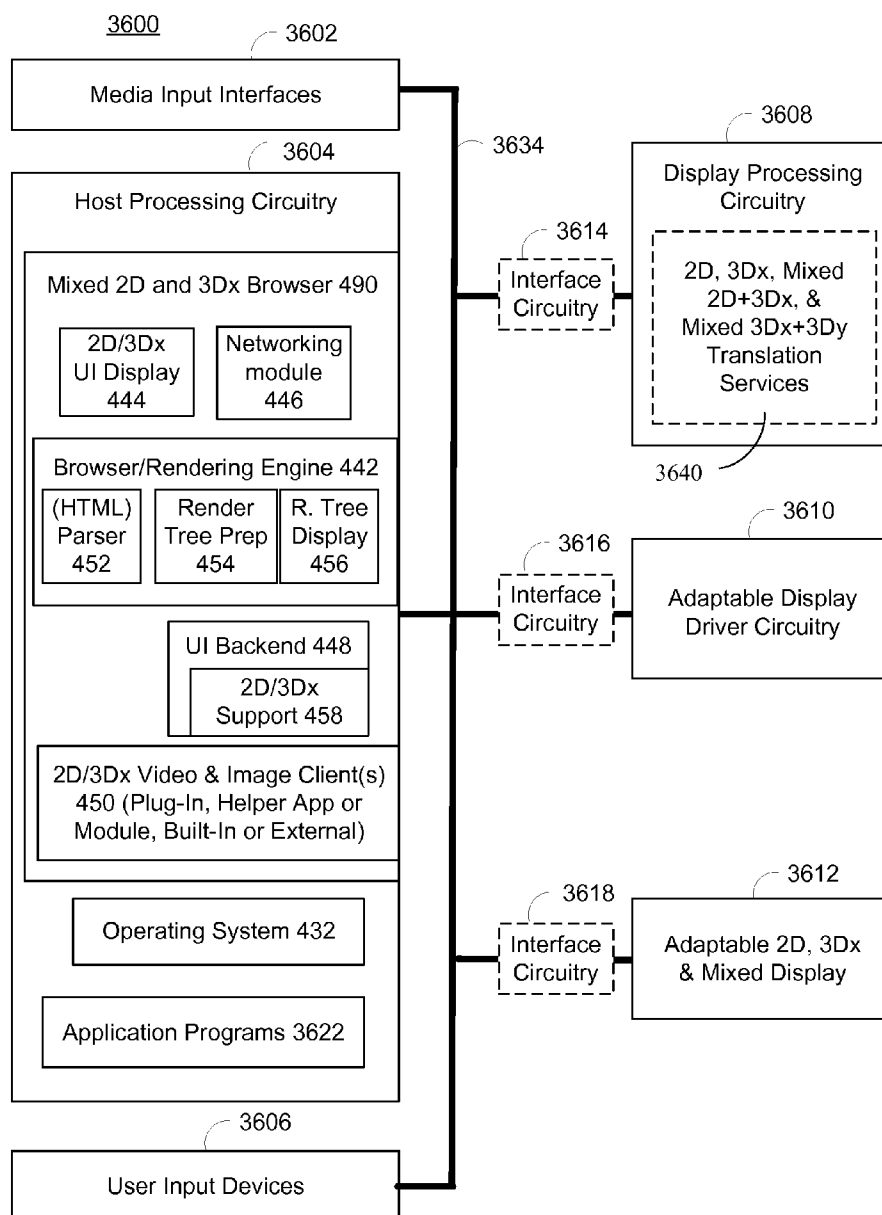


FIG. 36

INTERNET BROWSER AND ASSOCIATED CONTENT DEFINITION SUPPORTING MIXED TWO AND THREE DIMENSIONAL DISPLAYS

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the benefit of U.S. Provisional Application No. 61/291,818, filed on Dec. 31, 2009, which is incorporated by reference herein in its entirety; and

[0002] This application claims the benefit of U.S. Provisional Application No. 61/303,119, filed on Feb. 10, 2010, which is incorporated by reference herein in its entirety.

[0003] This application is also related to the following U.S. Patent Applications, each of which also claims the benefit of U.S. Provisional Patent Application Nos. 61/291,818 and 61/303,119 and each of which is incorporated by reference herein:

[0004] U.S. patent application Ser. No. 12/845,409, titled "Display With Adaptable Parallax Barrier," filed Jul. 28, 2010;

[0005] U.S. patent application Ser. No. 12/845,440, titled "Adaptable Parallax Barrier Supporting Mixed 2D And Stereoscopic 3D Display Regions," filed Jul. 28, 2010;

[0006] U.S. patent application Ser. No. 12/845,461, titled "Display Supporting Multiple Simultaneous 3D Views," filed Jul. 28, 2010;

[0007] U.S. patent application Ser. No. 12/774,307, titled "Display with Elastic Light Manipulator," filed May 5, 2010;

[0008] U.S. patent application Ser. No. _____, titled "Backlighting Array Supporting Adaptable Parallax Barrier," filed on same date herewith;

[0009] U.S. patent application Ser. No. _____, titled "Operating System Supporting Mixed 2D, Stereoscopic 3D And Multi-View 3D Displays," filed on same date herewith;

[0010] U.S. patent application Ser. No. _____, titled "Application Programming Interface Supporting Mixed Two And Three Dimensional Displays," filed on same date herewith;

[0011] U.S. patent application Ser. No. _____, titled "Programming Architecture Supporting Mixed Two And Three Dimensional Displays," filed on same date herewith; and

[0012] U.S. patent application Ser. No. _____, titled "Integrated Backlighting, Sub-Pixel and Display Driver Circuitry Supporting Adaptive 2D, Stereoscopic 3D and Multi-View 3D Displays," filed on same date herewith.

BACKGROUND OF THE INVENTION

[0013] 1. Field of the Invention

[0014] The present invention relates to web browsers.

[0015] 2. Background Art

[0016] Images may be generated for display in various forms. For instance, television (TV) is a widely used telecommunication medium for transmitting and displaying images in monochromatic ("black and white") or color form. Conventionally, images are provided in analog form and are displayed by display devices in two-dimensions. More recently, images are being provided in digital form for display in two-dimensions on display devices having improved resolution (e.g., "high definition" or "HD"). Even more recently, images capable of being displayed in three-dimensions are being generated.

[0017] Conventional displays may use a variety of techniques to achieve three-dimensional image viewing functionality. For example, various types of glasses have been developed that may be worn by users to view three-dimensional images displayed by a conventional display. Examples of such glasses include glasses that utilize color filters or polarized filters. In each case, the lenses of the glasses pass two-dimensional images of differing perspective to the user's left and right eyes. The images are combined in the visual center of the brain of the user to be perceived as a three-dimensional image. In another example, synchronized left eye, right eye LCD (liquid crystal display) shutter glasses may be used with conventional two-dimensional displays to create a three-dimensional viewing illusion. In still another example, LCD display glasses are being used to display three-dimensional images to a user. The lenses of the LCD display glasses include corresponding displays that provide images of differing perspective to the user's eyes, to be perceived by the user as three-dimensional.

[0018] Some displays are configured for viewing three-dimensional images without the user having to wear special glasses, such as by using techniques of autostereoscopy. For example, a display may include a parallax barrier that has a layer of material with a series of precision slits. The parallax barrier is placed proximal to a display so that a user's eyes each see a different set of pixels to create a sense of depth through parallax. Another type of display for viewing three-dimensional images is one that includes a lenticular lens. A lenticular lens includes an array of magnifying lenses configured so that when viewed from slightly different angles, different images are magnified. Displays are being developed that use lenticular lenses to enable autostereoscopic images to be generated.

[0019] As such, many types of display devices exist that are capable of displaying three-dimensional images, and further types are being developed. Different types of displays that enable three-dimensional image viewing may have different capabilities and attributes, including having different depth resolutions, being configured for three-dimensional image viewing only, being switchable between two-dimensional image viewing and three-dimensional image viewing, and further capabilities and attributes.

[0020] Web browsers are applications that enable the retrieving, presenting, and traversing of information resources that are available on the World Wide Web ("the Web"). Web browsers may be included in electronic devices such as desktop computers and handheld devices to enable users to interact with Web-based information resources. Examples of information resources that may be retrieved and presented by a web browser include web pages, images, and videos. Some of these information resources may include two-dimensional or three-dimensional content.

BRIEF SUMMARY OF THE INVENTION

[0021] Methods, systems, and apparatuses are described for a browser that enables display of network-accessible content by display devices that have two-dimensional and three-dimensional display capability, substantially as shown in and/or described herein in connection with at least one of the figures, as set forth more completely in the claims.

BRIEF DESCRIPTION OF THE DRAWINGS/FIGURES

[0022] The accompanying drawings, which are incorporated herein and form a part of the specification, illustrate the

present invention and, together with the description, further serve to explain the principles of the invention and to enable a person skilled in the pertinent art to make and use the invention.

[0023] FIG. 1 shows a block diagram of a system that includes a web browser that supports mixed 2D (two-dimensional) and 3D (three-dimensional) displays, according to an exemplary embodiment.

[0024] FIG. 2 shows a block diagram of a web browser that supports mixed 2D and 3D displays interfaced with various display devices, according to an exemplary embodiment.

[0025] FIG. 3 shows a block diagram of examples of the web browser of FIG. 1 transmitting commands to a display device, according to embodiments.

[0026] FIG. 4A shows a block diagram of an electronic device that includes a browser architecture that supports mixed 2D and 3D displays, according to an exemplary embodiment.

[0027] FIG. 4B shows a block diagram of a display system that includes a 2D and 3D display enabled-browser architecture, according to an embodiment.

[0028] FIG. 5 shows a flowchart providing a process for enabling the display of 2D and 3D content using a web browser, according to an exemplary embodiment.

[0029] FIG. 6 shows a block diagram of a web browser configuring a display device of display of 2D and 3D content, according to an exemplary embodiment.

[0030] FIG. 7 shows a flowchart providing a process for using tag information to configure a screen for the display of 2D and 3D content, according to an exemplary embodiment.

[0031] FIGS. 8, 9, 10A, and 10B show examples of a screen displaying mixed 2D and 3D content in various screen regions, including tabs, frames, and objects, according to embodiments.

[0032] FIG. 11 shows a block diagram of a rendering engine configured to translate 3D content to 2D content and to translate a first type of 3D content to a second type of 3D content, according to an exemplary embodiment.

[0033] FIG. 12 shows a flowchart providing a process for determining display screen characteristics, according to an exemplary embodiment.

[0034] FIG. 13 shows a block diagram of storage that stores browser preferences, according to an exemplary embodiment.

[0035] FIG. 14 shows a block diagram of a display device having a light manipulator that enables display of 3D content by a screen, according to an exemplary embodiment.

[0036] FIG. 15 shows a block diagram of a display device having an adaptable light manipulator that enables the adaptable display of 3D content by a screen, according to an exemplary embodiment.

[0037] FIGS. 16 and 17 show block diagrams of examples of the display device of FIG. 15, according to embodiments.

[0038] FIG. 18 shows a flowchart for generating three-dimensional images, according to an exemplary embodiment.

[0039] FIG. 19 shows a cross-sectional view of an example of a display system, according to an embodiment.

[0040] FIGS. 20 and 21 shows view of example parallax barriers with non-blocking slits, according to embodiments.

[0041] FIG. 22 shows a view of the barrier element array of FIG. 22 configured to enable the simultaneous display of two-dimensional and three-dimensional images of various sizes and shapes, according to an exemplary embodiment.

[0042] FIG. 23 shows a view of the parallax barrier of FIG. 22 with differently oriented non-blocking slits, according to an exemplary embodiment.

[0043] FIG. 24 shows a display system providing two two-dimensional images that are correspondingly viewable by a first viewer and a second viewer, according to an exemplary embodiment.

[0044] FIG. 25 shows a flowchart for generating multiple three-dimensional images, according to an exemplary embodiment.

[0045] FIG. 26 shows a cross-sectional view of an example of the display system of FIG. 15, according to an embodiment.

[0046] FIGS. 27 and 28 show views of a lenticular lens, according to an exemplary embodiment.

[0047] FIG. 29 shows a flowchart for generating multiple three-dimensional images using multiple light manipulator layers, according to an exemplary embodiment.

[0048] FIG. 30 shows a block diagram of a display system, according to an exemplary embodiment.

[0049] FIGS. 31 and 32 show cross-sectional views of a display system, according to an exemplary embodiment.

[0050] FIG. 33 shows a block diagram of a display system, according to an exemplary embodiment.

[0051] FIG. 34 shows a block diagram of a display environment, according to an exemplary embodiment.

[0052] FIG. 35 shows a block diagram of an example electronic device, according to an embodiment.

[0053] FIG. 36 shows a block diagram of a display system that supports mixed 2D, stereoscopic 3D and multi-view 3D displays, according to an exemplary embodiment.

[0054] The present invention will now be described with reference to the accompanying drawings. In the drawings, like reference numbers indicate identical or functionally similar elements. Additionally, the left-most digit(s) of a reference number identifies the drawing in which the reference number first appears.

DETAILED DESCRIPTION OF THE INVENTION

I. Introduction

[0055] The present specification discloses one or more embodiments that incorporate the features of the invention. The disclosed embodiment(s) merely exemplify various aspects of the invention. The scope of the invention is not limited to the disclosed embodiment(s). The invention is defined by the claims appended hereto.

[0056] References in the specification to “one embodiment,” “an embodiment,” “an example embodiment,” etc., indicate that the embodiment described may include a particular feature, structure, or characteristic, but every embodiment may not necessarily include the particular feature, structure, or characteristic. Moreover, such phrases are not necessarily referring to the same embodiment. Further, when a particular feature, structure, or characteristic is described in connection with an embodiment, it is submitted that it is within the knowledge of one skilled in the art to effect such feature, structure, or characteristic in connection with other embodiments whether or not explicitly described.

[0057] Furthermore, it should be understood that spatial descriptions (e.g., “above,” “below,” “up,” “left,” “right,” “down,” “top,” “bottom,” “vertical,” “horizontal,” etc.) used herein are for purposes of illustration only, and that practical

implementations of the structures described herein can be spatially arranged in any orientation or manner.

II. Exemplary Embodiments

[0058] Embodiments of the present invention relate to web browsers that enable the display of two- and three-dimensional content. For instance, such web browsers may be enabled to display web pages, images, video, content generated by browser scripts and applications, and further types of information resources that include 2D and/or 3D content. In one example, a browser may be capable of processing a markup language document that defines one or more browser windows, frames, or tabs, within which to display web pages, images, and/or video content. The markup language document may include elements (e.g., tags) that specify one or more parameters to be associated with the displayed regions and content. In further examples, the browser may determine parameters to be associated with displayed regions and content based on other factors, such as a type of content to be displayed, a filename for the content, configuration information stored at a media server for the content, etc. The web browsers may generate configuration commands based on the determined parameters that cause display screens to be configured to display the 2D and/or 3D content.

[0059] Numerous types of display devices may display 2D and 3D content provided by the web browsers. For example, the display devices may include one or more light manipulators, such as parallax barriers and/or lenticular lenses, to deliver 3D media content in the form of images or views to the eyes of the viewers. Other types may include display devices with 3D display pixel constructs that may or may not employ such light manipulators. When used, light manipulators may be fixed or dynamically modified to change the manner in which the views are delivered. For instance, embodiments enable light manipulators that are adaptable to accommodate a changing viewer sweet spot, switching between two-dimensional (2D), stereoscopic three-dimensional (3D), and multi-view 3D views, as well as the simultaneous display of 2D, stereoscopic 3D, and multi-view 3D content. With regard to parallax barriers, example features that may be dynamically modified include one or more of a number of slits in the parallax barriers, the dimensions of each slit, the spacing between the slits, and the orientation of the slits. Slits of the parallax barriers may also be turned on or off in relation to certain regions of the screen such that simultaneous mixed 2D, stereoscopic 3D, and multi-view 3D presentations can be accommodated. Similarly, a lenticular lens may be dynamically modified, such as by modifying a width of the lenticular lens, to modify delivered images.

[0060] The following subsections describe numerous exemplary embodiments of the present invention. For instance, the next subsection describes embodiments for web browsers, followed by a subsection that describes embodiments for displaying content using a browser, a subsection that describes user input interface and web browser start up embodiments, a subsection that describes example display environments, and a subsection that describes example electronic devices. It is noted that the section/subsection headings are not intended to be limiting. Embodiments are described throughout this document, and any type of embodiment may be included under any section/subsection.

[0061] It will be apparent to persons skilled in the relevant art that various changes in form and detail can be made to the embodiments described herein without departing from the

spirit and scope of the invention. Thus, the breadth and scope of the present invention should not be limited by any of exemplary embodiments described herein.

A. Example Web Browser Embodiments

[0062] In embodiments, web browsers that provide native support for the display of mixed content are provided. For example, in one embodiment, a web browser comprises a graphical user interface (GUI) in which video content can be displayed in a window of the browser. Furthermore, one or more parameters (e.g., indicated via “tags” or by other configuration information) may be associated with the browser window and/or the displayed content. The parameters can specify various display characteristics, such as one or more of: a type of video content to be displayed within the browser window (e.g., 2D, stereoscopic 3D, or a particular type of multi-view 3D), a desired orientation of the displayed video content, a brightness/contrast to be associated with the browser window, and/or a video resolution to be associated with the browser window. The parameters to be associated with a browser window may be specified programmatically or determined dynamically at run-time. The parameters may also be modified at run-time by a user through a user control interface provided by the web browser. The web browser is further configured to cause one or more function calls to be placed to a graphics API (application programming interface), operating system, or device driver so that a window is opened on the display and the content is presented therein in a manner that is consistent with the associated parameters.

[0063] For instance, FIG. 1 shows a block diagram of a system 100, according to an exemplary embodiment. As shown in FIG. 1, system 100 includes a display device 102, a document server 104, a web browser 106, and a network 116. System 100 is a system where web browser 106 interfaces together one or more users and network content with display device 102. System 100 is described as follows.

[0064] System 100 may be implemented in one or more devices. For example, in one embodiment, web browser 106 and display device 102 may be implemented in a common electronic device 112 that may be accessed by a user, such as a mobile computing device (e.g., a handheld computer, a laptop computer, a notebook computer, a tablet computer (e.g., an Apple iPad™), a netbook, etc.), a mobile phone (e.g., a cell phone, a smart phone), a mobile email device, some types of televisions, etc. In another embodiment, as shown in FIG. 1, web browser 106 may be implemented in an electronic device 110 that is separate from display device 102. For instance, device 110 may be a home theater system receiver, a set-top box, a computer, a game console, or other such device, and display device 102 may be a display device that is coupled to device 110 in a wired or wireless fashion.

[0065] Web browser 106, also referred to as an “Internet browser” or “browser,” is an application for retrieving, presenting, and traversing network-based information resources. For instance, web browser 106 may be implemented in software (e.g., computer programs and/or data) that runs on a device. Web browser 106 may load an external information resource identified by a Uniform Resource Locator (URL), such as a web page, an image, a video, or other item of content. Web browser 106 may display the loaded information in a window of the browser. An information resource loaded by web browser 106 may reference further information resources, which may be loaded by web browser 106 for display. An information resource may include hyperlinks that

when displayed can be selected by a user to enable the user to navigate to the related information resources.

[0066] For instance, as shown in FIG. 1, document server 104 may store one or more information resources, such as an information resource 114. Information resource 114 may be an XML document, an HTML document (e.g., a web page), an image file, a video, or other type of information resource. A user of browser 106 may desire to view information resource 114, and may interact with browser 106 to cause information resource 114 to be loaded by browser 106. For instance, the user may enter the URL of information resource 114 into browser 106, or may select a hyperlink in a markup document that links to information resource 114, to cause browser 106 to load information resource 114. In response to the user interaction with browser 106, browser 106 may generate request 118, such a HTTP (hypertext transfer protocol) request (e.g., if the URL starts with “http:” or “https:”) or other type of request (e.g., FTP (file transfer protocol), etc.), which is transmitted from the device that includes browser 106. Request 118 is directed to a location of information resource 114 according to the URL of information resource 114.

[0067] Request 118 may be transmitted through a network 116 to be received by document server 104. For instance, network 116 may be any type of communication network, including a local area network (LAN), a wide area network (WAN), or a combination of communication networks, such as the Internet. Document server 104 may be any suitable type of computer system capable of providing documents over a network, such as a server, etc.

[0068] In response to receiving request 118, document server 104 locates and identifies information resource 114, and transmits information resource 114 to browser 106 through network 116. Browser 106 receives information resource 114, and displays content of information resource 114 in a window in a screen of display device 102.

[0069] As shown in FIG. 1, web browser 106 includes mixed 2D/3D supporting logic 108. Mixed 2D/3D supporting logic 108 enables web browser 106 to support display of mixed 2D and 3D content, according to an exemplary embodiment. For example, logic 108 may enable web browser 106 to provide two- and three-dimensional content for display by display devices that are capable of separately displaying two-dimensional and three-dimensional content, display devices that are capable of simultaneously displaying two-dimensional and three-dimensional content, display devices that are capable of simultaneously displaying different types of three-dimensional content, as well as display devices that can adaptively change the display of two-dimensional and three-dimensional contents (e.g., by changing display screen regions).

[0070] For example, with regard to information resource 114, 2D/3D supporting logic 108 may be capable of enabling browser 106 to render 2D and 3D content at display device 102 in a manner based on the contents of information resource 114 and/or based on one or more tags (e.g., HTML tags) or other configuration information associated with information resource 114 in a markup document that refers to information resource 114. An HTML document is a type of markup document that includes a tree of HTML elements and other information (e.g., textual information, etc.) according to an HTML language format. Each HTML element can have attributes assigned. In HTML syntax, some elements may be written with associated tags to assign attributes to the elements. An

element may be written with a start tag and an end tag, with the content indicated in between the start tag and end tag. A start tag includes the name of the element surrounded by angle brackets, and the corresponding end tag includes a slash character followed by the name of the element, which are both surrounded by angle brackets (not all elements necessarily include an end tag). For instance, a paragraph may be indicated by a “p” element. An example of a p element is shown as follows:

[0071] <p>Paragraph text is included in here . . . </p>

Furthermore, one or more attributes may be specified in the start tag. An attribute is defined in a start tag with a name of the attribute, an equal sign, and a value of the attribute (which may or may not be in quotes). For instance, the “abbr” element, which represents an abbreviation, expects a “title” attribute with its expansion. An example of the abbr element is shown as follows:

[0072] <abbr title=“Hyper Text Markup Language”>HTML</abbr>

[0073] In embodiments, tags may be included in markup documents to indicate types and characteristics of 2D and 3D content included or referenced therein. For instance, a “3D” element may be defined to indicate particular content as three-dimensional. An example 3D element is shown as follows:

[0074] <3D>mediafile.mpg</3D>

where mediafile.mpg is an MPEG video file that is indicated by the 3D tag as containing 3D content. Similarly, a “2D” element may be defined to indicate particular content as two-dimensional. An example 2D element is shown as follows:

[0075] <2D>imagefile.jpg</2D>

where imagefile.jpg is an JPG image file that is indicated by the 2D tag as containing 2D content. Further elements/tags may be defined to indicate various types of three-dimensional content, such as “3D-4” to indicate 3D multiview content with four camera views, “3D-8” to indicate 3D multiview content with eight camera views, “3D-HVGA” to indicate three-dimensional HVGA (half-size VGA), etc. It is noted that other characters than “3D” and “2D” may be used to indicate content as three-dimensional or two-dimensional.

[0076] Furthermore, attributes may be added to a 3D element to indicate the various type of 3D content. For instance, a 3D-4 video file may be indicated by the 3D element as:

[0077] <3D multiviewtype=“4”>mediafile2.mpg</3D>

where the start 3D tag includes a “multiviewtype” attribute indicating a type of multiview 3D content. A 3D element may include any number of attributes to indicate one or more 3D display characteristics, such as stereoscopic depth (e.g., a “depth” attribute), brightness (e.g., a “brightness” attribute), a size of a region of a screen in which the 3D content is to be displayed (e.g., a “regionsize” attribute having parameters such as row, column, width, and height parameters), a resolution attribute (e.g., a “resolution” attribute), a window orientation attribute indicating whether image content is to be displayed vertically or rotated by 90 degrees or by other amount (e.g., an “orientation” attribute), a freeform window attribute indicating a non-rectangular shape for the displayed content (e.g., a “freeform” attribute), a display region type attribute (e.g., a frame, a tab, etc.), etc. Any of these and/or additional three-dimensional display characteristics may be defined by separate elements/tags, or as attributes of an element/tag. Furthermore, a single element may be used to determine whether content is two-dimensional or three-dimensional:

[0078] <content type="3D">image2file.jpg</2D>

where the element "content" has a "type" attribute which can have the value of "3D" to indicate 3D content, "2D" to indicate 2D content, and possibly further values (e.g., "3D-4" to indicate 3D-4 multiview content, etc.).

[0079] As such, configuration information for display of content may be extracted from a web page according to the HTML language format (e.g., tags, attributes, etc.). Such configuration information may be provided in the form of tags as indicated above, or in further ways, as would become apparent to persons skilled in the relevant art(s) from the teachings herein. Furthermore, as described in further detail below, configuration information for content may be determined in other ways, including being determined by the filename of the content (e.g., by file extension), by configuration information stored at the file server that serves the content, and/or in further ways.

[0080] In embodiments, display device 102 may be one of a variety of display devices capable of displaying two-dimensional and/or three-dimensional content. For instance, FIG. 2 shows a block diagram of a display system 200, which is an exemplary embodiment of system 100 of FIG. 1. As shown in FIG. 2, system 200 includes web browser 106, a first display device 202, and a second display device 204. As shown in FIG. 2, web browser 106 includes mixed 2D/3D supporting logic 108. First display device 202 is a display device that is only capable of displaying two-dimensional content, and second display device 204 is a display device that is capable of displaying two-dimensional content and three-dimensional content. In the example of FIG. 2, via mixed 2D/3D supporting logic 108, web browser 106 is capable of displaying content at first and second display devices 202 and 204. In embodiments, web browser 106 may be capable of providing content for display by first and second display devices 202 and 204 one at a time. In another embodiment, web browser 106 may be capable of providing content for display by first and second display devices 202 and 204 and/or other combinations and numbers of display devices simultaneously.

[0081] Note that web browser 106 may be interfaced with display devices in various ways. For instance, FIG. 3 shows a block diagram of web browser 106 interfacing with display device 102 of FIG. 1, according to embodiments. As shown in FIG. 3, browser 106 can be interfaced with display device 102 through an API (application programming interface) 302 and a display driver 306, through an operating system (OS) 304 and display driver 306, and/or through display driver 306. FIG. 3 is described as follows.

[0082] API 302 is an interface implemented in software (e.g., computer program code or logic) for applications such as browser 106 that enables the applications to interact with other software and/or hardware. API 302 may be configured to perform graphics operations on graphics information received from the applications. API 302 may be implemented in a same device as browser 106. API 302 may be a special purpose API, or may be a commercially available API, such as Microsoft DirectX® (e.g., Direct3D®), OpenGL®, or other 3D graphics API, which may be modified according to embodiments to receive commands and/or content from browser 106. Further description of implementations of API 302 and other API implementations described herein is provided in pending U.S. patent application Ser. No. _____, titled "Application Programming Interface Supporting Mixed

Two And Three Dimensional Displays," filed on same date herewith, which is incorporated by reference herein in its entirety.

[0083] OS 304 is configured to interface users and applications with hardware, such as display device 102. OS 304 may be a commercially available or proprietary operating system. For instance, OS 304 may be an operating system such as Microsoft Windows®, Apple Mac OS® X, Google Android™, or Linux®, which may be modified according to embodiments. Further description of implementations of OS 304 and other operating system implementations described herein is provided in pending U.S. patent application Ser. No. _____, titled "Operating System Supporting Mixed 2D, Stereoscopic 3D And Multi-View 3D Display," filed on same date herewith, which is incorporated by reference herein in its entirety.

[0084] Display driver 306 may be implemented in software, and enables applications (e.g., higher-level application code) such as browser 106 to interact with display device 102. Display driver 306 may be implemented in a same device as browser 106. Multiple display drivers 306 may be present, and each display driver 306 is typically display device-specific, although some display drivers 306 may be capable of driving multiple types of display devices. Each type of display device typically is controlled by its own display device-specific commands. In contrast, most applications communicate with display devices according to high-level device-generic commands. Display driver 306 accepts the generic high-level commands (directly from browser 106, or via API 302 and/or OS 304), and breaks them into a series of low-level display device-specific commands, as used by the particular display device.

[0085] As shown in FIG. 3, browser 106 may generate a command 308 associated with the display of 2D and/or 3D content that is received by API 302. API 302 passes command 308 to display driver 306 (in a modified or unmodified form). Display driver 306 receives command 308, and generates one or more control signals 314 received by display device 102. Control signal(s) 314 place(s) a screen of display device 102 in a display mode corresponding to command 308. Furthermore, browser 106 may stream content to display device 102 through API 302 and display driver 306 to be displayed in the screen configured according to command 308. Note the API 302 may be included in OS 304 (when present), or may be separate. Furthermore, API 302 may communicate directly with display driver 306 as shown in FIG. 3, or may communicate with display driver 306 through OS 304.

[0086] Alternatively, browser 106 may generate a command 310 associated with the display of 2D and/or 3D content that is received by OS 304. OS 304 passes command 310 to display driver 306 (in a modified or unmodified form). Display driver 306 receives command 310, and generates control signal(s) 314 received by display device 102. Control signal(s) 314 place(s) a screen of display device 102 in a display mode corresponding to command 310. Furthermore, browser 106 may stream content to display device 102 through OS 304 and display driver 306 to be displayed in the screen configured according to command 310.

[0087] In another example, browser 106 may generate a command 312 associated with the display of 2D and/or 3D content that is directly received by display driver 306 (e.g., does not pass through API 302 or OS 304). Display driver 306 receives command 312, and generates control signal(s) 314 received by display device 102. Control signal(s) 314 place(s)

a screen of display device **102** in a display mode corresponding to command **312**. Furthermore, browser **106** may stream content to display device **102** directly through display driver **306** to be displayed by the screen configured according to command **312**.

[0088] As such, commands and content may be provided by browser **106** to display device **102** through one or more different intermediate components, which may include one or more of API **302**, OS **304**, and display driver **306**. Note that in an embodiment, browser **106** may include one or more API **302** and/or OS **304**, or may be included in OS **304**, in a similar manner as some commercially available operating systems that incorporate a web browser (e.g., Google Chrome OS™).

[0089] Web browser **106** may be implemented in various ways to interface users and network-based content with display devices that are capable of displaying two-dimensional content and/or three-dimensional content. For instance, FIG. 4A shows a block diagram of an electronic device **412** that includes a browser architecture for a web browser **400**, according to an exemplary embodiment. Device **412** may be any of the electronic devices mentioned herein as including a web browser (e.g., electronic devices **110** and **112** of FIG. 1), or may be an alternative device. Browser **400** is configured to interface users and network-based content with display devices that are capable of displaying two-dimensional content and/or three-dimensional content. In an embodiment, browser **400** may be a proprietary web browser. In another embodiment, browser **400** may be a commercially available web browser that is modified to enable users and network-based content to be interface with display devices capable of displaying two-dimensional content and/or three-dimensional content. For instance, web browsers such as Internet Explorer®, developed by Microsoft Corp. of Redmond, Wash., Mozilla Firefox®, developed by Mozilla Corp. of Mountain View, Calif., or Google® Chrome of Mountain View, Calif. may be modified according to embodiments. As shown in FIG. 4A, browser **400** includes various browser portions, including a user interface **402**, a rendering engine **404**, one or more optional client applications **406**, a networking module **408**, and a code interpreter **410**. These features of browser **400** are described as follows.

[0090] User interface **402** is configured to display information to enable a person to interact with browser **400**. For instance, user interface **402** may provide one or more graphical user interface (GUI) control elements that a user may interact with to use and/or configure browser **400**. For instance, user interface **402** may provide an address bar into which a user may enter URLs of desired information resources, a back button, a forward button, a refresh button, a stop button, a home button, one or more additional/alternative buttons, one or more pull down menus (e.g., a list of bookmarks, etc.), etc.

[0091] Networking module **408** is a communications module for browser **400** with network-accessible entities, such as document server **104** shown in FIG. 1. For instance, networking module **408** may be configured to generate network calls, such as HTTP requests (e.g., request **118** of FIG. 1) and/or other types of requests. The calls may be transmitted over a network (e.g., network **116** of FIG. 1) to remote entities to retrieve information resources corresponding to a URL in an address bar provided by user interface **402**, a URL for an information resource referenced in a markup document loaded by browser **400**, or a hyperlink present in content displayed by browser **400**.

[0092] Rendering engine **404** is configured to display requested content in one or more browser windows. For example, rendering engine **404** may request and receive a markup document (also known as a “markup language document”), and may render the content included in or referenced by the markup document for display in a screen of a display device. Rendering engine **404** can render displays of HTML (hypertext markup language) and XML (extensible markup language) documents, as well as image/video content. In the case where the markup document is an XML or HTML document (e.g., a web page), rendering engine **404** may parse the document to generate a DOM (document object model) tree. The DOM is a cross-platform and language-independent convention for representing objects in HTML and XML documents. Rendering engine **404** may generate a render tree from the DOM tree. Rendering engine **404** may perform a layout process to determine screen coordinates for each node of the render tree, and may traverse and “paint” each node of the render tree on the display screen in a browser window.

[0093] One or more client applications **406** may optionally be present. Each client application **406** may be interfaced with web browser **400** to add corresponding capabilities to web browser **406**, if web browser **406** does not already have such capabilities. For example, a client application **406** may be a presentation component configured to enable web browser **400** to play video, to scan for viruses, to display additional file types, such as PDF (portable document format) files, etc. Examples of client application **406** may include a media player, an Adobe® Flash® plug-in that enables animation, video, and interactivity for web pages, a Apple QuickTime® plug-in that enables various formats of digital video, images, sound, and interactivity for web pages, a Microsoft® Silverlight™ plug-in that enables multimedia, graphics, animation, and interactivity for web pages, etc.

[0094] Code interpreter **410** (also known as a “script engine”) is configured to interpret and execute script code referenced by markup documents. For example, code interpreter **410** may be configured to interpret and execute JavaScript® code. JavaScript® may be present to provide enhanced user interfaces and dynamic web pages. Code interpreter **410** may interpret JavaScript® source code, and execute the interpreted code. Similarly to code interpreter **410**, web browser **400** may include a compiled code execution module that is capable of executing compiled code, such as Java bytecode. For instance, browser **400** may include a virtual machine configured as a Java runtime environment to run Java applets, which may provide interactive features to web pages, including complex graphics.

[0095] As described above, browser **400** of FIG. 4A is configured to interface users and network-accessible resources with display devices that are capable of displaying two-dimensional content and/or three-dimensional content. Browser **400** may be configured in various ways to perform its functions, and various embodiments for browser **400** are described herein. For instance, FIG. 4B shows a block diagram of a display system **480** that includes a 2D and 3D display enabled-browser architecture, according to an embodiment. As shown in FIG. 4B, display system **480** includes a web browser **490**. Browser **490** is an embodiment of browser **400** that is configured to interface users and network-accessible resources with display devices that are capable of displaying two-dimensional content and/or three-dimensional content. The embodiment of browser **490** shown in FIG. 4B is provided for purposes of illustration, and is not

intended to be limiting. In further embodiments, browser **490** may include fewer, additional, and/or alternative features than shown in FIG. 4B.

[0096] Display system **480** is an example of a display system that is capable of displaying mixed 2D and 3D content (e.g., via mixed 2D/3D supporting logic **108**). As shown in FIG. 4B, system **480** includes web browser **490**, operating system kernel and kernel utilities with regional/3Dx support **432** (“OS **432**”), one or more browser page and 2D/3Dx content servers **460** (“server **460**”), first-third display circuitry **416a-416c**, a 2D display **418a**, a 3D display with 2D mode **418b**, a regionally configurable 2D/3Dx display **418c**, and a network **478**. Web browser **490** includes various browser portions, including a browser/rendering engine **442**, a 2D/3Dx UI (user interface) display **444**, a networking module **446**, a UI backend **448**, and one or more 2D/3Dx video and image client(s) **450**. Browser/rendering engine **442** includes a parser **452**, a render tree preparation module **454**, and a rendered tree display **456**. UI backend **448** includes 2D/3Dx support **458**. Browser page and 2D/3Dx content server(s) **460** includes page content **462**, linked content file or files **464**, and a streaming server application **466**. Page content **462** includes a hypertext content link **468**, a screen region location **470**, and an underlying screen configuration **472**. Linked content file or files **464** includes a file A and screen configuration A **474**, and a file B and screen configuration B **476**. OS **432** includes user input interfaces **420**, a 2D, 3Dx & mixed display driver interface **422**, shell operations **424**, 2D, 3Dx, mixed 2D and 3Dx, & mixed 3Dx and 3Dy translation services **426**, an API supporting regional 2D/3Dx **428** (“API **428**”), and one or more communication interfaces **440**. 2D, 3Dx and mixed display driver interface **422** includes 2D only driver variant **434**, 3Dx only driver variant **436**, and mixed 2D and 3Ds driver variant **438**. First-third display circuitry **416a-416c** each includes a corresponding one of translation services **430a-430c**. The features of system **480** are described as follows.

[0097] 2D display **418a**, 3D display with 2D mode **418b**, and regionally configurable 2D/3Dx display **418c** are example types of display devices that may display content provided by browser **490**. One or more of displays **418a-418c** may be present. 2D display **418a** is an example of 2D display device **202** of FIG. 2, and is a display device that is only capable of displaying two-dimensional content. 3Dx display with 2D mode **418b** is an example of 2D-3D display device **204** of FIG. 2, and is a display device that is capable of displaying two-dimensional and three-dimensional content. For instance, 3Dx display with 2D mode **418b** may be set in a 2D mode where 3Dx display with 2D mode **418b** can display 2D content in full screen, but not 3D content, and may be set in a 3D mode where 3Dx display with 2D mode **418b** can display 3D content in full screen, but not 2D content. Furthermore, 3Dx display with 2D mode **418b** may be capable of displaying 3D content having multiple camera views (“multiview”)—a number of “x” views—such as 3D-4, having four camera views, 3D-16, having sixteen camera views, etc. The additional camera views enable viewers to “view behind” displayed 3D content by moving their heads left-right, as further described elsewhere herein. Regionally configurable 2D/3Dx display **418c** is an example of 2D-3D display device **204** of FIG. 2, and is a display device that is capable of displaying two-dimensional and three-dimensional content simultaneously. For instance, regionally configurable 2D/3Dx display **418c** may display 2D content in one

or more regions of a display screen while simultaneously displaying 3D content in one or more other regions of the display screen. Furthermore, regionally configurable 2D/3Dx display **418c** may be capable of displaying 3D content having multiple camera views.

[0098] Network **478** is an example of network **116** in FIG. 1, and browser page and 2D/3Dx content server **460** is an example of document server **104** in FIG. 1. One or more browser page and 2D/3Dx content servers **460** may be present that are accessible to browser **490** over network **478**. Browser page and 2D/3Dx content server **460** may include one or more information resources, such as markup documents (e.g., web pages, etc.), image files, video files, etc. For example, page content **462** is an example of markup document content. Page content **462** may include text, page configuration information, references to other information resources, etc. For instance, as shown in FIG. 4B, page content **462** may include one or more hypertext content links **468**, which are links displayed in a page generated from page content **462** and displayed by browser **490**. Hypertext content link **468** may be selected by a user to traverse to and display an information resource as a page element. Screen region location **470** may be present to indicate a region in the displayed page in which a page element corresponding to hypertext content link **468** is to be displayed. For example, screen region location **470** may be used by a layout module of browser/rendering engine **442** to select a location for display of the corresponding content in a display screen. Underlying screen configuration **472** may be present to indicate a screen display configuration for the displayed page, including desired 2D and/or 3D display characteristics of the screen. For example, underlying screen configuration **472** may be included in a file that includes page content **462** (e.g., in the form of one or more tags), or may be separately stored in server **460**. Browser/rendering engine **442** may use information of underlying screen configuration **472** in a configuration request to configure a screen region for displaying the corresponding content.

[0099] Linked content file(s) **464** includes files that may be requested for display by browser **490** (e.g., in request **118**), such as in response to a user selecting a hyperlink in a displayed page. In some cases, linked content file(s) **464** may include multiple files from which a file may be selected to be provided in a response to a user selecting a hyperlink. For instance, as shown in FIG. 4B, linked content file(s) **464** may include file A and screen configuration A **474**, and a file B and screen configuration B **476**. File A and file B are alternative files to be provided to browser **490** in response to a request. File A corresponds to a screen configuration A, and file B corresponds to a different screen configuration B. For instance, file A or file B may be provided by server **460** in response to a request based on characteristics of a display screen in which content of the file is to be displayed, based on a provided display frame size, based on communication link characteristics (testing), and/or based on other criteria.

[0100] Streaming server application **466** may be present in browser page and 2D/3Dx content server **460** to stream video content in response to a request from browser **490** to server **460** for video files.

[0101] OS **432** is an example of operating system **304** shown in FIG. 3. OS **432** interfaces applications, such as browser **490**, with displays **418a-418c**. As indicated in FIG. 4B, OS **432** may provide various forms of 2D/3Dx display support. For instance, API supporting regional 2D/3Dx **428** is configured to interface one or more applications (e.g.,

browser 490) with OS 432, and thereby interface the applications with a display device (e.g., one or more of displays 418a-418c) coupled to OS 432. API supporting regional 2D/3Dx 428 is configured to enable applications, such as browser 490, to access various display functions, including enabling regional definition for 2D, 3D, and 3Dx content displayed by display screens and further display functions.

[0102] User input interfaces 420 are configured to receive user input to enable a person to interact with display system 480, browser 490, and content displayed by displays 418a-418c. Further example embodiments for user input interfaces 420 are described elsewhere herein.

[0103] 2D, 3Dx & mixed display driver interface 422 enables applications, such as browser 490, that interface with OS 432 via API 428 to provide and control two- and/or three-dimensional content displayed at a displays 418a-418c. 2D only driver variant 434, 3Dx only driver variant 436, and mixed 2D and 3Dx driver variant 438 are examples of display driver 306 of FIG. 3. 2D, 3Dx & mixed display driver interface 422 may forward commands (e.g., from browser 490) to 2D only driver variant 434 when 2D display 418a is present, enabling only 2D-related commands to be processed. 2D, 3Dx & mixed display driver interface 422 may forward commands to 3Dx only driver variant 436 when 3Dx display with 2D mode 418b is present, enabling 2D or 3Dx related commands to be processed. 2D, 3Dx & mixed display driver interface 422 may forward commands to mixed 2D and 3Dx driver variant 438 when regionally configurable 2D/3Dx display 418c is present, enabling regional 2D or 3Dx related commands to be processed.

[0104] Shell operations 424 may be present in OS 432 to control and/or enable user configuration of environmental properties, such as the 2D and/or 3D display configuration of an environmental background, of desktop icons, of displayed windows, etc. In embodiments, shell operations 424 may be implemented in hardware, software, firmware, or any combination thereof, including as a shell operations module.

[0105] Mixed 2D and 3Dx, & mixed 3Dx and 3Dy translation services 426 may be present in OS 432 to provide for translation of received content (e.g., from an application such as browser 490) from a first dimensionality to a second dimensionality. For instance, translation services 426 may be configured to translate received 3D content to 2D content, such as when an application provides 3D content, and 2D display 418a is the target display (e.g., the target display is not capable of displaying 3D content). In another example, translation services 426 may be configured to translate a first type of 3D content to a second type of 3D content, such as when an application provides regional 2D and/or 3D content, and 3Dx display with 2D mode is the target display (e.g., the target display is not capable of displaying content regionally), and/or to translate 3D content having a first number “x” of cameras (e.g., 3D-8 content) to 3D content having a second number “y” of cameras (e.g., 3D-4 content), if the target display does not support “x” camera views. Still further, translation services 426 may be configured to translate 2D content to 3D content, and/or may be able to perform other forms of content translations. Example embodiments for mixed 2D and 3Dx, & mixed 3Dx and 3Dy translation services 426 (e.g., translators) are described elsewhere herein.

[0106] Further description regarding an operating system configured to interface applications with displays supporting two-dimensional and three-dimensional views, such as OS 432, is provided in pending U.S. patent application Ser. No.

_____, titled “Operating System Supporting Mixed 2D, Stereoscopic 3D And Multi-View 3D Displays,” which is incorporated by reference herein in its entirety.

[0107] Display circuitry 416a-416c may have the form of hardware, software, firmware, or any combination thereof, such as having the form of a graphics card, circuitry etc. Display circuitry 416a-416c may be present to interface OS 432 with displays 418a-418c, respectively. Display circuitry 416a-416c may receive content signals and control signals from OS 432, and may be configured to generate drive signals to drive displays 418a-418c, respectively. Examples of display circuitry (e.g., drive circuits) are described elsewhere herein.

[0108] As shown in FIG. 4B, display circuitry 416a-416c may each optionally include a corresponding one of translation services 430a-430c. When present, translation services 430a-430c may perform translations of received content in a similar manner as mixed 2D and 3Dx, & mixed 3Dx and 3Dy translation services 426. For instance, translation services 430a may translate received 3D content to 2D content for display by 2D display 418a. Translation services 430b may translate received regionally configurable 2D and/or 3D content to non-regional 2D and/or 3D content for display by 3Dx display with 2D mode display 418b. Translation services 430b and 430c may each translate unsupported types of 3D content to supported types of 3D content for display by 3Dx display with 2D mode display 418b and regionally configurable 2D/3Dx display 418c, respectively. Translation services 430a-430c may also be configured to perform additional and/or alternative forms of content translations, in embodiments.

[0109] Browser 490 is configured to enable network-accessible content to be displayed in two- and three-dimensions at displays 418a-418c. 2D/3Dx UI display 444 is an example of user interface 402 shown in FIG. 4A. 2D/3Dx UI display 444 may include an address bar, back/forward buttons, bookmarking, and/or further portions the browser display (e.g., other than the main window displaying a requested page). 2D/3Dx UI display 444 may include 2D & 3Dx counterparts, such as images or video streams (e.g., 2D/3Dx Applet like functionality).

[0110] Browser/Rendering Engine 442 is an example of rendering engine 404 of FIG. 4A. Engine 442 processes HTML, and manages the display of web page and 2D and 3D image & Video (stream) file content. For instance, parser 452 may parse a loaded HTML document to generate a DOM (document object model) tree, as described above. Rendering tree preparation module 454 may generate a render tree from the DOM tree. Module 454 may identify screen configurations to be applied to regions of the display screen based on the render tree, and may cause configuration requests to be generated based on the identified screen configuration to cause a configuration or reconfiguration of the screen in the regions. Module 454 may include a layout module that performs a layout process to determine screen coordinates for each node of the render tree. Module 454 may traverse and “paint” each node of the render tree in a browser window on the display screen, to generate render tree display 456.

[0111] Networking module 446 is an example of networking module 408 shown in FIG. 4A. In an embodiment, networking module 446 is platform independent, and interfaces with OS 432 to operate through communication interface(s) 440 of OS 432 via network protocols (e.g., HTTP requests, etc.).

[0112] UI Backend 448 is configured to draw basic widgets, such as drop down boxes, combo boxes, and windows. UI Backend 448 may interface with API 428 of OS 432 to generate 2D and 3D image or video (e.g., streamed) elements. UI Backend 448 may be platform independent.

[0113] 2D/3Dx video and image client(s) 450 are an example of client application(s) 406 of FIG. 4A. 2D/3Dx video and image client(s) 450 may include plug-ins, add-ons, built-in, external helper apps, etc., that provide functionality to browser 480. In one mode of operation, clients 450 may provide the functionality for: (i) generating the control signals that are passed to OS 432 for configuring display screen regions in preparation for underlying video/image presentation; (ii) managing the retrieval of media content to be displayed; (iii) delivering the media content via OS 432 to one or more of displays 418a-418c; and (iv) managing the presentation of such media content (e.g., enabling rewind, zoom, pause, etc.). Alternatively, item (i) above may be performed by browser/renderer engine 442 according to HTML tag definitions, for example. Similarly, one or more others of (ii)-(iv) may be performed by engine 442. In an embodiment, client(s) 450 can be integrated into engine 442, or may remain a plug-in, an add-on, a built-in, a helper app as shown in FIG. 4B. Client(s) 450 may reside outside of browser 490, and launching and loading of an external client 450 may be performed by browser 490 within another external window. Note that code interpreters (e.g., code interpreter 410 of FIG. 4A), such as a Java interpreter, may also be present in browser 490, which operate pursuant to code of client 450 to perform a same function as a compiled add-on.

[0114] The embodiments of display system 480 and browser 490 shown in FIG. 4B are provided for purposes of illustration. In further embodiments, display system 480 and browser 490 may include fewer, further, and/or alternative components, as would be known to persons skilled in the relevant art(s). Further embodiments regarding the features of display system 480 and browsers 400 and 490 are described in the following subsections.

B. Exemplary Embodiments for Displaying Content Using a Browser

[0115] As described above, browser 400 may retrieve two-dimensional and three-dimensional content for display by display devices, including content associated with a web page. FIG. 5 shows a flowchart 500 providing a process for displaying web page content, according to an exemplary embodiment. Flowchart 500 may be performed by browsers described elsewhere herein, such as browser 400 of FIG. 4A or browser 490 of FIG. 4B. Flowchart 500 is described with respect to FIG. 6, which shows a block diagram of browser 400 interfaced with a display device 606, according to an exemplary embodiment. In the example of FIG. 6, browser 400 includes rendering engine 404, application client(s) 406, and code interpreter 410. Furthermore, rendering engine 404 includes mixed 2D/3D supporting logic 108. Display driver 604 is an example of display driver 306 of FIG. 3. Device 412 of FIG. 4A is not shown in FIG. 6 for ease of illustration, but it is noted that browser 400 may be included in device 412, and display device 606 may be included in or may be external to device 412. Further structural and operational embodiments will be apparent to persons skilled in the relevant art(s) based on the discussion regarding flowchart 500. Flowchart 500 is described as follows.

[0116] Flowchart 500 begins with step 502. In step 502, web page content is parsed. For example, as shown in FIG. 6, browser 400 may receive a markup document 608. Markup document 608 may include HTML text that describes a web page. Rendering engine 404 may parse markup document 608. For instance, rendering engine 404 may include parser 452 of FIG. 4B, which may be configured to parse HTML documents, such as markup document 608. Parser 452 may receive the content of markup document 608 in 8K chunks or portions, and may begin parsing the underlying HTML text of markup document 608 on a chunk-by-chunk basis without waiting for all content to be received. Alternatively, parser 452 may receive all of the content of markup document 608 before beginning parsing. Parser 452 may generate a document object model tree or other structure that identifies each of the elements of content included in or referenced by markup document 608. A first portion of the elements may relate to two-dimensional content, and second portion of the elements may relate to three-dimensional content. Alternatively, all of the elements may relate to three-dimensional content (or two-dimensional content).

[0117] In step 504, two-dimensional content to be displayed in a first region of the screen is identified. For example, rendering module 404 may identify a first object of markup document 608 that relates to two-dimensional content. The first object may be identified by parser 452 encountering a hypertext link corresponding to the first object in markup document 608, or in other manner. The first object may include any form of two-dimensional content, such as an image, a video, another web page, etc. The first object may be identified in various ways. Upon identification, rendering engine 404 may handle further processing of the first object, client application 406 may be selected to manage the processing of the first object (e.g., for a particular type of first object that the client application 406 is configured to process), or code interpreter 410 may interpret and execute the first object when the first object is an un-compiled script.

[0118] For instance, in an embodiment, the first object may be identified based on an identifier for the first object (e.g., a filename) or a structure of the first object itself (e.g., file contents, such as header information). For example, a MIME (multipurpose Internet mail extensions) type file extension to a filename for the first object provided in markup document 608 may be used to identify the first object, to identify that the first object includes 2D content, and to select rendering engine 404 or a particular client application 406 to process the first object. In another embodiment, the first object may be identified by a content server (e.g., content server 460 of FIG. 4B) from which the first object is requested. For instance, one or more parameters, such as tag information or other information, may be present in markup document 608 that may be passed to the content server by rendering engine 404 in a request that can be used to select the first object to be returned in response to the request. For instance, the tag(s) and/or other parameter(s) may indicate a screen configuration for a screen 620 of display device 606, a frame size to be generated by rendering engine 404, and/or other information. The content server may use the tag(s) and/or other parameters to select the first object, and/or may use other information to select the first object, such as characteristics of the communication link between the file server and browser. Referring to FIG. 4B, file A or file B may be selected by server 460 based on whether the browser screen configuration matches screen configuration A or screen configuration B stored at server 460, and the

selected file is transmitted (e.g., an image file is transmitted, video is streamed, etc.) to browser 400 (e.g., as information resource 610 in FIG. 6).

[0119] In step 506, a first configuration request is communicated to at least attempt to cause a first configuration of the first region of the screen to support the two-dimensional content. For example, as shown in FIG. 6, rendering engine 404 (e.g., render tree preparation module 454 of FIG. 4B) may generate a command 612 that is a configuration request for a first region of screen 620 to support display of the 2D content identified in step 504. For instance, as described above with respect to FIG. 3, command 612 may be transmitted from rendering engine 404 directly, or through an API and/or OS, to display driver 604. Display driver 604 receives command 612, and generates control signal(s) 616 that are received by display device 606. Control signal(s) 616 place(s) a region of screen 620 in a 2D display mode for display of the identified 2D content.

[0120] In step 508, three-dimensional content to be displayed in a second region of the screen is identified. For example, in a similar fashion as described above with respect to step 504, rendering module 404 may identify a second object of markup document 608 that relates to three-dimensional content. The second object may be identified by parser 452 encountering a hypertext link corresponding to the second object in markup document 608, or in other manner. The second object may include any form of three-dimensional content, such as an image, a video, another web page, etc., and any type of three-dimensional content (e.g., stereoscopic 3D, 3D-2, 3D-4, etc.). The second object may be identified in various ways. Upon identification, rendering engine 404 may handle further processing of the second object, a client application 406 may be selected to manage the processing of the second object (e.g., for a particular type of second object that the client application 406 is configured to process), or code interpreter 410 may interpret and execute a script of the second object.

[0121] For instance, in a similar manner as described above, the second object may be identified based on an identifier for the second object (e.g., a filename) or a structure of the second object itself (e.g., file contents, such as header information). For example, a MIME (multipurpose Internet mail extensions) type file extension to a filename for the second object provided in markup document 608 may be used to identify the second object, to identify that the second object includes 3D content, and to select rendering engine 404 or a particular client application 406 to process the second object. In another embodiment, the second object may be identified by a content server (e.g., content server 460 of FIG. 4B) from which the second object is requested. For instance, one or more parameters, such as tag information or other information, may be present in markup document 608 that may be passed to the content server by rendering engine 404 in a request that can be used to select the second object to be returned in response to the request. For instance, the tag(s) and/or other parameter(s) may indicate a screen configuration for a screen 620 of display device 606, a frame size to be generated by rendering engine 404, and/or other information. The content server may use the tag(s) and/or other parameters to select the second object, and/or may use other information to select the second object, such as characteristics of the communication link between the file server and browser. Referring to FIG. 4B, file A or file B may be selected by server 460 based on whether the browser screen configuration

matches screen configuration A or screen configuration B stored at server 460, and the selected file is transmitted (e.g., an image file is transmitted, video is streamed, etc.) to browser 400 (e.g., as information resource 610 in FIG. 6).

[0122] In step 510, a second configuration request is communicated to at least attempt to cause a second configuration of the second region of the screen to support the three-dimensional content, the first configuration being different from the second configuration. For example, as shown in FIG. 6, rendering engine 404 (e.g., render tree preparation module 454 of FIG. 4B) may generate a second command 614 that is a second configuration request for a second region of screen 620 to support display of the 3D content identified in step 508. For instance, command 614 may be transmitted from rendering engine 404 directly, or through an API and/or OS, to display driver 604. Display driver 604 receives command 614, and generates control signal(s) 618 that are received by display device 606. Control signal(s) 618 place(s) a second region of screen 620 in a 3D display mode for display of the identified 3D content. If the second region of screen 620 is in a different display mode (e.g., in a 2D display mode, or a different 3D display mode), the second region of the screen 620 is reconfigured according to the second configuration request.

[0123] To provide the content to the first and second regions of screen 620, rendering engine 404 may generate a render tree for each of the 2D and 3D content identified in steps 504 and 506, and may perform a layout process to determine screen coordinates (positional information) for each node of each render tree (e.g., using render tree preparation module 454 shown in FIG. 4B). Rendering engine 404 may traverse each node of each render tree for display on screen 620, and may generate graphical data representative of each render tree to paint each node. As shown in FIG. 6, rendering engine 404 may transmit 2D graphical data 622 corresponding to the identified 2D content, and 3D graphical data 624 corresponding to the identified 3D content. Display driver 604 may receive 2D graphical data 622 and 3D graphical data 624, and transmit corresponding processed 2D graphical data 626 and processed 3D graphical data 628 that are received by display device 606. Display device 606 may display the 2D content of processed 2D graphical data 626 in the first region of screen 620, which is configured according to the first configuration request. Furthermore, display device 606 may display the 3D content of processed 3D graphical data 628 in the second region of screen 620, which is configured according to the second configuration request. In this manner, browser 400 enables simultaneously display of 2D and 3D content by a display screen.

[0124] As described above, tags may be included in markup document 608. The tags may be used to define characteristics of the display of 2D and 3D content by a display device. For instance, the tags may be used to indicate one or more display properties of the displayed content, including indicating whether content is 2D or 3D, indicating a type of 3D content, etc. FIG. 7 shows a flowchart 700 providing a process for using tags to configure the display of 2D and 3D content, according to an exemplary embodiment. Flowchart 700 may be performed by browser embodiments described herein, such as browser 400 of FIG. 4A or browser 490 of FIG. 4B. Flowchart 700 is described with respect to FIG. 6 for purposes of illustration. Further structural and operational embodi-

ments will be apparent to persons skilled in the relevant art(s) based on the discussion regarding flowchart 700. Flowchart 700 is described as follows.

[0125] Flowchart 700 begins with step 702. In step 702, first tag information associated with two-dimensional content is identified, the two-dimensional content intended for both a left eye and a right eye of a viewer. For example, rendering module 404 of FIG. 6 may identify a first tagged object in markup document 608 that relates to two-dimensional content. As described above for two-dimensional content, same images are delivered to the right and left eyes of a viewer so that the content is perceived as two-dimensional. The first tagged object may be identified by parser 452 encountering a URL or other content identifier (e.g., a filename) that has associated tags in markup document 608, or in other manner. The first tagged object may include any form of two-dimensional content, such as an image, a video, another web page, etc. The tag information associated with the two-dimensional content may include any number of attributes. The tag information may indicate a screen configuration for screen 620 of display device 606, a frame size to be generated by rendering engine 404 for display of the 2D content, a type of the 2D content, a display brightness for the 2D content, a resolution for the 2D content (e.g., 720p, 1080p, etc.), and/or any other suitable information described elsewhere herein or otherwise known.

[0126] In step 704, second tag information associated with three-dimensional content is identified, the three-dimensional content having a first portion and a second portion, the first portion intended for the left eye of the viewer and the second portion intended for the right eye of the viewer, the first portion being a first camera view and the second portion being a second camera view. For example, rendering module 404 of FIG. 6 may identify a second tagged object in markup document 608 that relates to three-dimensional content. As described above for three-dimensional content, images of differing perspective are delivered to the right and left eyes of a viewer. The images are combined in the visual center of the brain of the viewer to be perceived as a three-dimensional image. The second tagged object may be identified by parser 452 encountering a second URL or other content identifier (e.g., a filename) that has associated tags in markup document 608, or in other manner. The second tagged object may include any form of three-dimensional content, such as an image, a video, another web page, etc. The second tag information associated with the three-dimensional content may include any number of attributes. The second tag information may indicate a screen configuration for screen 620 of display device 606 for display of the 3D content, a frame size to be generated by rendering engine 404 for display of the 3D content, a type of the 3D content, a display brightness for the 3D content, a display resolution for the 3D content, and/or any other suitable information described elsewhere herein or otherwise known.

[0127] In step 706, the presentation of the two-dimensional content is caused in a first region of a screen. For instance, rendering engine 404 may generate command 612 that is a configuration request for a first region of screen 620 to support display of the 2D content according to the first tag information identified in step 702. As described above with respect to FIG. 3, command 612 may be transmitted from rendering engine 404 directly, or through an API and/or OS, to display driver 604. Display driver 604 receives command 612, and generates control signal(s) 616 that are received by display

device 606. Control signal(s) 616 place(s) a first region of screen 620 in a 2D display mode for display of the 2D content.

[0128] Rendering engine 404 may generate a render tree for the 2D content, and may perform a layout process to determine screen coordinates (positional information) for each node of the render tree (e.g., using render tree preparation module 454 shown in FIG. 4B). Rendering engine 404 may traverse each node of the render tree, and may generate graphical data representative of the render tree to paint each node. As shown in FIG. 6, rendering engine 404 may transmit 2D graphical data 622 corresponding to the 2D content. Display driver 604 may receive 2D graphical data 622 and transmit corresponding processed 2D graphical data 626 that is received by display device 606. Display device 606 may display the 2D content of processed 2D graphical data 626 in the first region of screen 620, which is configured according to the first configuration request.

[0129] In step 708, the presentation of the three-dimensional content is caused in a second region of the screen. For instance, rendering engine 404 may generate command 614 that is a configuration request for a second region of screen 620 to support display of the 3D content according to the second tag information identified in step 704. Command 614 may be transmitted from rendering engine 404 directly, or through an API and/or OS, to display driver 604. Display driver 604 receives command 614, and generates control signal(s) 618 that are received by display device 606. Control signal(s) 618 place(s) a second region of screen 620 in a 2D display mode for display of the 2D content.

[0130] Rendering engine 404 may generate a render tree for the 3D content, and may perform a layout process to determine screen coordinates for each node of the render tree. Rendering engine 404 may traverse each node of the render tree, and may generate graphical data representative of the render tree to paint each node. As shown in FIG. 6, rendering engine 404 may transmit 3D graphical data 624 corresponding to the 3D content. Display driver 604 may receive 3D graphical data 624 and transmit corresponding processed 3D graphical data 628 that is received by display device 606. Display device 606 may display the 3D content of processed 3D graphical data 628 in the second region of screen 620, which is configured according to the second configuration request. In this manner, browser 400 causes display of the 3D content on screen 620 of display screen 606 simultaneously with the display of the 2D content on screen 620.

[0131] As such, according to flowcharts 500 and 700, two-dimensional and three-dimensional content identified by browser 400 may simultaneously be displayed within corresponding regions of screen 620. Furthermore, different types of three-dimensional content (e.g., different resolutions, different numbers of image pairs, different stereoscopic depths, etc.) are enabled to be individually or simultaneously displayed by browser 400. In embodiments, any number of different types of two-dimensional and three-dimensional content may be displayed in any number of regions of screen 620.

[0132] For instance, FIGS. 8, 9, 10A, and 10B show examples of screen 620 displaying content in various screen regions, including tabs, frames, and display objects, according to embodiments. FIG. 8 shows screen 620 of FIG. 6 displaying a browser window 802 that includes multiple frames. Frames enable browsers to display two or more web pages or other media elements within the same browser window (e.g., side-by-side, etc.). Frames may be defined using “frameset” tags that define frames and their sizes. FIG. 8

shows browser window **802** including a first frame **804** and a second frame **806**. First frame **804** is configured for the display of two-dimensional content (e.g., according to step **506** of FIG. **5** or step **706** of FIG. **7**), and second frame **806** is configured for the display of three-dimensional content (e.g., according to step **510** of FIG. **5** or step **708** of FIG. **7**). In the example of FIG. **8**, first frame **804** and second frame **806** have approximately the same size, and are positioned side-by-side. In other embodiments, first and second frames **804** and **806** may have different sizes, and may have different positions relative to each other (e.g., above and below, etc.). Still further, although first and second frames **804** and **806** are shown as having rectangular shapes in FIG. **8**, in other embodiments, first and second frames **804** and **806** may have other shapes. Note that any number of frames may be displayed in browser window **802** that respectively display two-dimensional or three-dimensional content.

[0133] In another example, FIG. **9** shows screen **620** displaying a browser window **902** that includes multiple tabs. Tabs enable browsers to display two or more documents in a same browser window one at a time. The tabs can be used as a navigational widget to switch the display of the documents. FIG. **9** shows browser window **902** including a first tab region **904** and a second tab region **906**. First tab region **904** may be configured for the display of two-dimensional content (e.g., according to step **506** of FIG. **5** or step **706** of FIG. **7**), and second tab region **906** may be configured for the display of three-dimensional content (e.g., according to step **508** of FIG. **5** or step **708** of FIG. **7**). As shown in FIG. **9**, first and second tab regions **904** and **906** each have a corresponding tab extending upward that may be used to bring the respective region forward. First tab region **904** is displayed over second tab region **906**, such that second tab region **906** is not visible (except for the tab of second tab region **906**). The tab of second tab region **906** may be selected (e.g., by mouse click, etc.) to bring second tab region **906** to the forefront to be displayed over first tab region **904**, causing first tab region **904** to not be visible (except for the tab of first tab region **904**). Note that any number of tab regions may be present in browser window **902** that respectively display two-dimensional or three-dimensional content.

[0134] In another example, FIG. **10A** shows screen **620** displaying browser window **902** of FIG. **9**, with browser window **902** including tab regions **902** and **904**. First tab region **902** is displayed over second tab region **904**, and a frame **1002** is displayed in tab region **902**. Any number of frames may be displayed in a tab region. Furthermore, an object **1004** is displayed that overlaps first tab region **904** and frame **1002**. Object **1004** may be a two-dimensional object (e.g., displayed according to step **506** of FIG. **5** or step **706** of FIG. **7**) or a three-dimensional object (e.g., displayed according to step **510** of FIG. **5** or step **708** of FIG. **7**). In an embodiment, display of object **1004** may be a graphical object generated at least in part by client application **406** interacting with rendering engine **406**. For instance, object **1004** may be generated based on a Flash® application, a Java applet, etc., that is executed by client application **406** (or by rendering engine **404**). Note that any number of two-dimensional and/or three-dimensional content objects similar to object **1004** may be displayed in browser window **902**. Furthermore, although object **1004** is shown as having a round shape in FIG. **10A**, in other embodiments, object **1004** may

have other shapes (e.g., rectangular, other polygonal shape, shape of a person, an animal, an animated character, a product, etc.).

[0135] FIG. **10B** shows another example of screen **620** displaying a browser window **1020** similar to browser window **902** of FIG. **10A**, with browser window **1020** including tab regions **902** and **904**, and with tab region **904** including first frame **1002** and a second frame **1006**. Frames **1002** and **1006** may each include two-dimensional content (e.g., displayed according to step **506** of FIG. **5** or step **706** of FIG. **7**) or three-dimensional content (e.g., displayed according to step **510** of FIG. **5** or step **708** of FIG. **7**). Any number of two-dimensional and/or three-dimensional content objects similar to frames **100** and **1006** may be displayed in browser window **1020** having any shape.

[0136] Browser window **1020** includes various user interface elements providing controls for navigating the display of 2D and 3D content. As shown in FIG. **10B**, browser window **1020** (and any other browser windows described herein) may include a navigation bar **1008**, which may include various controls. A user may interact with navigation bar **1008** to navigate to web pages by entering corresponding URLs in an address entry box. Such web pages may include 2D and/or 3D content for display in browser window **1020**. A user may interact with back and forward buttons in navigation bar **1008** to navigate to a previous resource or forward to a subsequent resource. A user may interact with a refresh button of navigation bar **1008** to reload a current resource, and may interact with a stop button of navigation bar **1008** to cancel loading a resource. The example of navigation bar **1008** shown in FIG. **10B** is provided for purposes of illustration and is not intended to be limiting. In further embodiments, navigation bar **1008** may include additional and/or alternative navigation elements, such as a search engine query entry box, a home button, etc.

[0137] Furthermore, as shown in FIG. **10B**, browser window **1020** provides various browser controls for controlling the display of two-dimensional and three-dimensional content. For instance, as shown in FIG. **10B**, browser window **1020** may include a 3D display control bar **1010**. In the example of FIG. **10B**, 3D display control bar **1010** is positioned in a North position in browser window **1020** immediately below navigation bar **1008**, but in other embodiments may have other forms or positions (e.g., right side, left side, South position, etc.), and may be combined with other displayed bars. Furthermore, in other embodiments, 3D display control bar **1010** may have other forms, such as a widget, an icon, or other user interface element.

[0138] 3D display control bar **1010** enables a user to configure 3D display settings and/or preferences for browser window **1020**. For instance, 3D display control bar **1010** may include a 2D-3D toggle button **1014** and/or a 3D options button **1016**. 2D-3D toggle button **1014** may be selected (e.g., by clicking with a mouse pointer **1024**, by keystrokes, etc.) by a user to toggle between display of content in browser window **1020** in 2D form, or to enable 3D-enabled content to be displayed in 3D form. 2D-3D toggle button **1014** may display the current 2D-3D setting (e.g., either 2D or 3D). 3D options button **1016** may be selected by a user to set one or more 3D display settings/preferences for browser window **1020**. For example, in an embodiment, a user may select 3D options button **1016** to invoke a menu **1018** that lists one or more 3D display options that may be selected by the user. In the example of FIG. **10B**, menu **1018** includes a “set 3Dx” option

(to select a 3D multiview display type), a set 3D intensity option (to set a 3D display depth), a linked defaults option, and an advertisements defaults option. The linked defaults option enables a user to configure whether content invoked by clicking on a hyperlink in a web page displayed in browser window **1020** is displayed in 2D or 3D form. For instance, a user can set as a default all content generated by the same domain to be displayed regionally in full. Any hypertext linked content (e.g., coming from another source) may be set according to the linked defaults option be reduced to 2D or to be enabled to be displayed in 3D (e.g., of a particular 3D type). Thereafter, by clicking on content that has been reduced to 2D form, a restoration to 3D form may be performed. Content that was restored to full 3D may be clicked again to be reduced back to 2D form. A user may use the advertisements defaults option to set whether advertisements are displayed in 2D form by default, or whether 3D-enabled advertisements may be displayed in 3D form. For instance, an advertiser may attempt to push strong 3D effect graphics/video/text to users of browser window **1020** to grab their attention. This may be overridden through setup with the advertisements defaults option, or through direct user interaction with the advertisement itself. For example, a right click on the advertisement may generate reduce intensity/2D/3D/stop-pause" type options.

[0139] The example of 3D display control bar **1010** shown in FIG. **10B** is provided for purposes of illustration and is not intended to be limiting. In further embodiments, 3D display control bar **1010** may have other form or position, and may include additional and/or alternative 3D control elements.

[0140] Furthermore, tab regions may enable users to configure 2D-3D settings on a tab region-by-tab region basis. For instance, as shown in FIG. **10B**, tab region **904** may include a 3D user interface element **1012** that enables 3D settings to be made for tab region **1012**. 3D user interface element **1012** may be a button, icon, widget, may invoke a menu, etc., that a user may interact with to configure 2D-3D settings for tab region **904**. Such settings may be similar to those described above with respect to 3D display control bar **1010** and/or may include further and/or alternative settings. 3D user interface element **1012** is shown for purposes of illustration, and may have other form and capabilities than described with respect FIG. **10B**.

[0141] Still further, browser window **1020** may enable frames and/or specific content items to enable users to configure 2D-3D settings on frame-by-frame or content-by-content basis. For instance, as shown in FIG. **10B**, a user may invoke a menu **1022** with respect to frame **1002** (e.g., by right clicking pointer **1024** in frame **1002**) that provides one or more 2D-3D configuration options. For instance, as shown in FIG. **10B**, menu **1022** may include a toggle 2D-3D option (to toggle between display of content in 2D or 3D), a change 3Dx option (to change a 3D multiview display setting), an increase 3D intensity option, a reduce 3D intensity option, a pause option (to pause display of video content), etc. Menu **1022** is shown for purposes of illustration, and may provide further and/or alternative 2D-3D display related options to those shown in FIG. **10B**.

[0142] As shown in FIG. **10B**, browser window **1020** may include a 3D status bar **1012**. In the example of FIG. **10B**, 3D status bar **1012** is positioned in a South-most position in browser window **1020**, but in other embodiments may have other positions, and may be combined with other displayed bars. Furthermore, in other embodiments, 3D status bar **1012**

may have other forms, such as a widget, an icon, or other user interface element. 3D status bar **1012** displays a current 2D-3D setting status for browser window **1020**, and may optionally change the displayed 2D-3D setting status depending on the particular region (e.g., tab region, frame, content, etc.) over which pointer **1024** is hovered. 3D status bar **1012** may show any suitable 3D status information, such as whether display of 2D or 3D content is enabled, a type of 3D multiview that is displayed (e.g., "3D-8"), 2D-3D settings for advertisements, a 3D intensity setting, and/or further display information. The example of 3D status bar **1012** shown in FIG. **10B** is provided for purposes of illustration and is not intended to be limiting. In further embodiments, 3D status bar **1012** may have other form or position, and may include additional and/or alternative 3D status elements.

[0143] It is noted that the examples of FIGS. **8**, **9**, **10A**, and **10B** are provided for purposes of illustration, and are not intended to be limiting. In the examples of FIGS. **8**, **9**, **10A**, and **10B**, it is assumed that display device **606** supports the display of both two-dimensional and three-dimensional content. However, it is noted that not all types of display device **606** may support both two-dimensional content and three-dimensional content. Furthermore, not all types of display device **606** that support three-dimensional content may support all types of three-dimensional content. As such, in embodiments, browser **400** may be configured to translate unsupported types of content to supported types of content. Additionally and/or alternatively, browser **400** may be interfaced with components that are configured to perform such translations. For instance, as shown in FIG. **4B**, OS **432** includes translation services **426**, and display circuitry **416a-416c** include respective translation services **430a-430c**.

[0144] In an embodiment, rendering engine **404** of browser **400** may be configured to translate types of content that are not supported by a display device to supported types of content. FIG. **11** shows a block diagram of rendering engine **404**, according to an exemplary embodiment. As shown in FIG. **11**, rendering engine **404** includes a first translator **1102** and a second translator **1104**. In embodiments, rendering engine **404** may include one or both of first and second translators **1102** and **1104**. First translator **1102** may be present in rendering engine **404** to support display devices that do not support the display of three-dimensional content. Second translator **1102** may be present in rendering engine **404** to support display devices that do not support the display of one or more types of three-dimensional content.

[0145] First translator **1102** is configured to translate received 3D data to 2D data for display by a display device. For example, as shown in FIG. **11**, three-dimensional graphical data associated with an information resource may be received by rendering engine **404**. Rendering engine **404** may determine that the information resource contains three-dimensional content in any manner, included such as by a MIME file extension, by contents of a media file containing the data, by a tag associated with the information resource, etc. When a display device does not support the display of three-dimensional content, first translator **1102** may translate three-dimensional graphical data **1106** of the information resource to two-dimensional graphical data **1108**. Two-dimensional graphical data **1108** may be transmitted to the display device (e.g., 2D display of FIG. **4B**) to enable two-dimensional content to be displayed in a screen region based on two-dimensional graphical data **1106**.

[0146] Furthermore, a display device that supports the display of three-dimensional data may not support all types of three-dimensional data (e.g., the display device does not support 3D graphics data having additional camera views other than initial first right and left views, does not support a number of camera views greater than 3D-4, etc.). Second translator 1104 is configured to translate 3D data of an information resource of one or more unsupported 3D content types to 3D data of one or more supported 3D content types for display by a display device. For example, as shown in FIG. 11, first-type three-dimensional graphical data 1110 associated with an information resource may be received. Rendering engine 404 may determine that the first-type of three-dimensional content is an unsupported type in any manner, included such as by a MIME file extension, by contents of a media file containing the data, by a tag associated with the information resource, etc. When a display device does not support the display of the first-type of three-dimensional content, second translator 1104 may translate first-type three-dimensional graphical data 1110 to second-type three-dimensional graphical data 1112. Second-type three-dimensional graphical data 1112 is transmitted to display device 606 to enable the corresponding second type of three-dimensional content to be displayed in the region of screen 620.

[0147] First translator 1102 may be configured in various ways to translate received 3D data to 2D data. For instance, in an embodiment, three-dimensional graphical data 1106 may be received as a stream of right image data and left image data. First translator 1102 may be configured to combine the right and left image data into two-dimensional image data that defines a stream of two-dimensional images that may be output as two-dimensional data 1108. In another embodiment, first translator 1102 may be configured to select the right image data or the left image data to be output as two-dimensional data 1108, while the other of the right image data or left image data is not used. In further embodiments, first translator 1102 may translate received 3D data to 2D data in other ways.

[0148] Second translator 1104 may be configured in various ways to translate 3D data of a first 3D content type to 3D data of a second 3D content type. For instance, second translator 1104 may translate a first 3D multiview type (e.g., 3D-16) to a second 3D multiview type (e.g., 3D-4) or to a single 3D view. In such an embodiment, second translator 1104 may not pass extra left-right image pairs from first-type three-dimensional data 1110 to second-type three-dimensional data 1112. In an embodiment, second translator 1104 (and/or first translator 1102) may use techniques of image scaling to modify an unsupported display resolution to a supported display resolution. For instance, second translator 1104 may use upsampling or interpolating to increase resolution, and may use subsampling or downsampling to decrease resolution. In further embodiments, second translator 1104 may translate 3D data in other ways. In still further embodiments, a translator may be present to translate 2D content to 3D content, such as when a user has a preference to view content as 3D content. Various techniques may be used to convert 2D graphical data to 3D graphical data, as would be known to person skilled in the relevant art(s).

[0149] Note that a determination may be made of whether a display device supports the display of two-dimensional content and/or three-dimensional content, and/or a determination may be made of further display device characteristics in various ways. For instance, FIG. 12 shows a flowchart 1200

providing a process for determining display screen characteristics, according to an exemplary embodiment. Flowchart 1200 may be performed by browser 400 of FIG. 4A, browser 490 of FIG. 4B, etc.

[0150] In step 1202, an indication of at least one characteristic of the screen is requested. For instance, browser 400 of FIG. 6 may transmit a screen characteristic request to display device 606. The screen characteristic request may be transmitted through an API (e.g., API 302 of FIG. 3), an OS (e.g., OS 304 of FIG. 3), and/or a display driver (e.g., display driver 306 of FIG. 3), when present in a communication path between browser 400 and display device 606.

[0151] In step 1204, a response to the request is received. For example, display driver 606 may transmit a response to the screen characteristic request that includes an indication of one or more characteristics of screen 620, including whether screen 620 supports display of 2D and/or 3D content, an indication of supported types of 3D content, an indication of a resolution of screen 620, whether screen 620 supports display of mixed 2D and 3D, etc. The response may be transmitted through the display driver, OS, and/or API, when present. Browser 400 may receive the response, and rendering engine 404 may use the received response information to render 2D and/or 3D content that is supported by screen 620. For instance, due to the information included in the response, first translator 1102 or second translator 1104 may be activated to translate an unsupported content type to a supported content type, and/or other actions may be taken.

B. Example User Input Interface and Browser Start Up Embodiments

[0152] As described above, user input interfaces 420 in FIG. 4B receive user input to enable persons to interact with browser content displayed by a display device. For example, via user input interface 420, a user may be enabled to interact with displayed controls of browser 402 (e.g., displayed in 2D/3Dx UI display 444), to select tabs to view different tab regions, to interact with displayed graphical items (e.g., windows, frames, objects etc.), to modify (e.g., rotate, resize, etc.) displayed graphical items, etc. As described above, user interface 402 (FIG. 4A) of browser 400 may provide a command-line interface (e.g., a URL address entry box), a GUI, and/or other browser interface with which the user can interact using user input interfaces 420. In embodiments, user input interface 420 may enable users to interact with displayed controls of browser 400 to adjust three-dimensional characteristics of three-dimensional content displayed by browser 400 (e.g., rendered by rendering engine 404). For example, user input interface 420 may enable three-dimensionality of displayed content to be turned on or off (e.g., to toggle between two-dimensionality and three-dimensionality). User input interface 420 may enable a degree of three-dimensionality of displayed content to be modified (e.g., increased or decreased, such as by changing a depth of three-dimensionality, increasing or decreasing a number of supplied camera views, etc.), may enable three-dimensional objects to be rotated in three-dimensions, and/or may enable further types of adjustment to three-dimensional characteristics of displayed three-dimensional content. Furthermore, user input interface 420 may enable other characteristics of displayed content to be modified, such as modifying contrast, brightness, etc.

[0153] In embodiments, the user may interact with user input interface 402 in various ways, including using a mouse/

pointing device to move a displayed pointer/cursor. The pointer may be used to select control settings. The pointer may be used to “click and drag” objects to move them, to resize objects, to rotate objects, to select controls/settings, to open a pop-up menu, etc. In other embodiments, the user may interact with a keyboard, a thumb wheel or other wheel, a roller ball, a stick pointer, a touch sensitive display, any number of virtual interface elements (e.g., such as a keyboard or other user interface element displayed by screen 620), a voice recognition system, and/or other user interface elements described elsewhere herein or otherwise known to provide user input. For instance, user input interface 402 may support a touch screen that is reactive to user finger touches to the screen to cause three-dimensional characteristics of displayed objects to be modified. For instance, particular motions of one or more figures against the screen may cause object resizing, 3D rotation, movement in 3D, etc. (e.g., touching two fingers to the screen, and dragging them together may be interpreted as “grabbing” a window and moving the window in 3D).

[0154] In embodiments, users may have preferences with regard to a browser environment upon the browser being activated. Such preferences may include preferences with regard to display of three-dimensional content. For example, a user may desire for a browser to power up in a two-dimensional or three-dimensional display mode, and if a three-dimensional display mode is desired, the user may have particular three-dimensional display preferences (e.g., a preferred degree of displayed three-dimensionality). For instance, the user may desire for the various controls of the browser to be displayed in two- or three-dimensions, may desire all content to be displayed as two-dimensional or three-dimensional by default, may desire particular contents such as advertisements to be displayed as two-dimensional by default, etc.

[0155] Embodiments enable display preferences to be set by users, and to be used to configure the display environments of users upon device boot up, user login, browser activation, etc. For instance, FIG. 13 shows a block diagram of storage 1302 that may be included in an electronic device (e.g., device 412 of FIG. 4A) that includes browser 400, according to an exemplary embodiment. As shown in FIG. 13, storage 1302 stores user browser preferences 1304. User browser preferences 1304 may indicate the user preferences that a user may have for a browser environment upon the browser being activated, including the browser preferences mentioned above and/or further preferences. User preferences 1304 may be loaded at browser startup, and used (e.g., by rendering engine 404, OS 304 or 432, etc.) to enable the browser environment to be displayed as desired by a user. Storage 1402 may include one or more non-volatile storage elements, such as non-volatile random access memory (RAM) devices (e.g., flash memory, electrically erasable programmable read-only memory, etc.), read only memory (ROM) devices, a hard disk drive, a CDROM (compact disc ROM), a DVD (digital video disc), etc. User preferences 1304 may be associated with a user by being stored in a user account of the user, being stored in a cookie associated with the user, etc.

C. Example Display Device Screen Embodiments

[0156] Embodiments described herein for browsers that support the display of two-dimensional and three-dimensional content may be implemented with respect to various types of display devices. For example, as described above,

some display screens are configured for displaying two-dimensional content, although they may display two-dimensional images that may be combined to form three-dimensional images by special glasses worn by users. Some other types of display screens are capable of display two-dimensional content and three-dimensional content without the users having to wear special glasses using techniques of autostereoscopy. As described above, browser embodiments described herein may generate configuration requests/commands to configure regions of the display screen for display of content, and may provide the content for display in the configured regions. Display drivers (e.g., display driver 306 of FIG. 3, driver variants 434, 436, and 438 of FIG. 4B, etc.) may receive the configuration requests/commands, and may generate control signals to cause the screen to be configured as indicated. Furthermore, the display drivers may supply the content provided by the browsers to the display devices to be displayed on the screen. Example display devices, screens, and display drivers are described as follows that receive the control signals, are configured accordingly, and that receive and display the provided content.

[0157] As described above, display devices, such as display device 606, may be implemented in various ways. For instance, display device 606 may be a television display (e.g., an LCD (liquid crystal display) television, a plasma television, etc.), a computer monitor, or any other type of display device. Display device 606 may include any suitable type or combination of light and image generating devices, including an LCD screen, a plasma screen, an LED (light emitting device) screen (e.g., an OLED (organic LED) screen), etc. Furthermore, display device 606 may include any suitable type of light filtering device, such as a parallax barrier (e.g., an LCD filter, a mechanical filter (e.g., that incorporates individually controllable shutters), etc.) and/or a lenticular lens, and may be configured in any manner, including as a thin-film device (e.g., formed of a stack of thin film layers), etc. Furthermore, display device 606 may include any suitable light emitting device as backlighting, including a panel of LEDs or other light emitting elements.

[0158] For instance, FIG. 14 shows a block diagram of a display device 1400, according to an exemplary embodiment. As shown in FIG. 14, display device 1400 includes a screen 1402. Display device 1400 is an example of display device 606 and screen 1402 is an example of screen 620 described above (e.g., with respect to FIG. 6). Device 1400 receives one or more control signals 1406 (e.g., from browser 400) that are configured to place screen 620 in a desired display mode (e.g., either a two-dimensional display mode or a three-dimensional display mode). As shown in FIG. 14, screen 1404 includes a light manipulator 1404. Light manipulator 1404 is configured to manipulate light that passes through light manipulator 1404 to enable three-dimensional images to be delivered to users in a viewing space. For instance, control signal(s) 1406 may be configured to activate or deactivate light manipulator 1404 to place screen 620 in a three-dimensional display mode or a two-dimensional display mode, respectively.

[0159] Examples of light manipulator 1404 include a parallax barrier and a lenticular lens. For instance, light manipulator 1404 may be a parallax barrier that has a layer of material with a series of precision slits. The parallax barrier is placed proximal to a light emitting pixel array so that a user's eyes each see a different set of pixels to create a sense of depth through parallax. In another embodiment, light manipulator

1404 may be a lenticular lens that includes an array of magnifying lenses configured so that when viewed from slightly different angles, different images are magnified. Such a lenticular lens may be used to deliver light from a different set of pixels of a pixel array to each of the user's eyes to create a sense of depth. Embodiments are applicable display devices that include such light manipulators, include other types of light manipulators, and that may include multiple light manipulators.

[0160] As shown in FIG. 14, display device **1400** receives a content signal **1408** (e.g., from device **412** of FIG. 4A, or other electronic device). Content signal **1408** includes two-dimensional or three-dimensional content for display by screen **1402**, depending on the particular display mode. In the embodiment of FIG. 14, light manipulator **1404** is physically fixed—is not adaptable. As such, when present, light manipulator **1404** (e.g., a fixed parallax barrier or a fixed lenticular lens) always delivers three-dimensional images of a particular type to a particular region in a viewing space. As such, light manipulator **1404** is not adaptable to deliver other types of three-dimensional images and/or to deliver two and/or three-dimensional images to multiple different regions of a viewing space.

[0161] In contrast, FIG. 15 shows a block diagram of a display device **1500** that is adaptable, according to an exemplary embodiment. As shown in FIG. 15, display device **1502** includes a screen **1502**. Display device **1500** is an example of display device **606** and screen **1502** is an example of screen **620** described above (e.g., with respect to FIG. 6). Furthermore, as shown in FIG. 15, screen **1504** includes an adaptable light manipulator **1504**. Adaptable light manipulator **1504** is configured to manipulate light that passes through adaptable light manipulator **1504** to enable three-dimensional images to be delivered to users in a viewing space. Furthermore, adaptable light manipulator **1504** is adaptable—is not physically fixed in configuration. As such, adaptable light manipulator **1504** is adaptable to deliver multiple different types of three-dimensional images and/or to deliver three-dimensional images to different/moving regions of a viewing space. Furthermore, in an embodiment, different regions of adaptable light manipulator **1504** may be adaptable such that multiple two-dimensional and/or three-dimensional images may be simultaneously delivered by screen **1502** to the viewing space.

[0162] Device **1500** receives one or more control signals **1506** (e.g., from browser **400**) that are configured to place screen **1502** in a desired display mode (e.g., either a two-dimensional display mode or a three-dimensional display mode), and/or to configure three-dimensional characteristics of any number and type as described above, such as configuring adaptable light manipulator **1504** to deliver different types of three-dimensional images, to deliver three-dimensional images to different/moving regions of a viewing space, and to deliver two-dimensional and/or three-dimensional images from any number of regions of screen **1502** to the viewing space.

[0163] As shown in FIG. 15 display device **1500** receives a content signal **1508** (e.g., from device **412** of FIG. 4A, or other electronic device). Content signal **1508** includes two-dimensional and/or three-dimensional content for display by screen **1502**, depending on the particular display mode and on the number of regions of screen **1502** that are delivering different two- or three-dimensional views to a viewing space.

[0164] Content signals **1408** and **1508** may include video content according to any suitable format. For example, content signals **1408** and **1508** may include video content delivered over an HDMI (High-Definition Multimedia Interface) interface, over a coaxial cable, as composite video, as S-Video, a VGA (video graphics array) interface, etc. Note that control signals **1406** and **1506** may be provided separately or in a same signal stream to display devices as their corresponding one of content signals **1408** and **1508**.

[0165] Exemplary embodiments for display devices **1400** and **1500** of FIGS. 14 and 15 are described as follows for purposes of illustration.

1. Exemplary Embodiments Using Parallax Barriers

[0166] Display devices **1400** and **1500** may include parallax barriers as light manipulators **1404** and **1504**, respectively. For instance, FIG. 16 shows a block diagram of a display system **1600**, which is an example of display device **606**, according to an embodiment. As shown in FIG. 16, system **1600** includes a display device driver circuit **1602**, an image generator **1612**, and parallax barrier **1620**. As shown in FIG. 16, image generator **1612** includes a pixel array **1608**, and parallax barrier **1620** includes a barrier element array **1610**. Furthermore, as shown in FIG. 16, display driver circuit **1602** includes a pixel array driver circuit **1604** and a barrier array driver circuit **1606**. These features of system **1600** are described as follows.

[0167] Pixel array **1608** includes a two-dimensional array of pixels (e.g., arranged in a grid or other distribution). Pixel array **1608** is a self-illuminating or light-generating pixel array such that the pixels of pixel array **1608** each emit light included in light **1652** emitted from image generator **1612**. Each pixel may be a separately addressable light source (e.g., a pixel of a plasma display, an LCD display, an LED display such as an OLED display, or of other type of display). Each pixel of pixel array **1608** may be individually controllable to vary color and intensity. In an embodiment, each pixel of pixel array **1608** may include a plurality of sub-pixels that correspond to separate color channels, such as a trio of red, green, and blue sub-pixels included in each pixel.

[0168] Parallax barrier **1620** is positioned proximate to a surface of pixel array **1608**. Barrier element array **1610** is a layer of parallax barrier **1620** that includes a plurality of barrier elements or blocking regions arranged in an array. Each barrier element of the array is configured to be selectively opaque or transparent. Combinations of barrier elements may be configured to be selectively opaque or transparent to enable various effects. For example, in one embodiment, each barrier element may have a round, square, or rectangular shape, and barrier element array **1610** may have any number of rows of barrier elements that extend a vertical length of barrier element array **1610**. In another embodiment, each barrier element may have a “band” shape that extends a vertical length of barrier element array **1610**, such that barrier element array **1610** includes a single horizontal row of barrier elements. Each barrier element may include one or more of such bands, and different regions of barrier element array may include barrier elements that include different numbers of such bands.

[0169] One advantage of such a configuration where barrier elements extend a vertical length of barrier element array **1610** is that such barrier elements do not need to have spacing between them because there is no need for drive signal routing in such space. For instance, in a two-dimensional LCD

array configuration, such as TFT (thin film transistor) display, a transistor-plus-capacitor circuit is typically placed onsite at the corner of a single pixel in the array, and drive signals for such transistors are routed between the LCD pixels (row-column control, for example). In a pixel configuration for a parallax barrier, local transistor control may not be necessary because barrier elements may not need to be changing as rapidly as display pixels (e.g., pixels of pixel array 1608). For a single row of vertical bands of barrier elements, drive signals may be routed to the top and/or bottom of barrier elements. Because in such a configuration drive signal routing between rows is not needed, the vertical bands can be arranged side-by-side with little-to-no space in between. Thus, if the vertical bands are thin and oriented edge-to-edge, one band or multiple adjacent bands (e.g., five bands) may comprise a barrier element in a blocking state, followed by one band or multiple adjacent bands (e.g., two bands) that comprise a barrier element in a non-blocking state (a slit), and so on. In the example of five bands in a blocking state and two bands in a non-blocking state, the five bands may combine to offer a single black barrier element of approximately 2.5 times the width of a single transparent slit with no spaces therein.

[0170] It is noted that in some embodiments, barrier elements may be capable of being completely transparent or opaque, and in other embodiments, barrier elements may not be capable of being fully transparent or opaque. For instance, such barrier elements may be capable of being 95% transparent when considered to be “transparent” and may be capable of being 5% transparent when considered to be “opaque.” “Transparent” and “opaque” as used herein are intended to encompass barrier elements being substantially transparent (e.g., greater than 75% transparent, including completely transparent) and substantially opaque (e.g., less than 25% transparent, including completely opaque), respectively.

[0171] Display driver circuit 1602 receives control signal 1622 and content signal 1624. As described below, content signal 1624 includes two-dimensional and/or three-dimensional content for display. Control signal 1622 may be control signal 1406 of FIG. 14 (for a non-adaptable parallax barrier 1620) or may be control signal 1506 of FIG. 15 (for an adaptable parallax barrier 1620). Control signal 1622 may be received from a display driver of an operating system (e.g., may be control signal 618 received from display driver 604 in FIG. 6). Display driver circuit 1602 is configured to generate drive signals based on control signal 1622 and content signal 1624 to enable display system 1600 to display two-dimensional and three-dimensional images to users 1618 in viewing space 1670. For example, pixel array driver circuit 1604 is configured to generate a drive signal 1614 that is received by pixel array 1608 (e.g., based on content signal 1624 and/or control signal 1622). Drive signal 1614 may include one or more drive signals used to cause pixels of pixel array 1608 to emit light 1652 of particular desired colors and/or intensity. Barrier array driver circuit 1606 is configured to generate a drive signal 1616 that is received by barrier element array 1610 (e.g., based on control signal 1622). Drive signal 1616 may include one or more drive signals used to cause each of the barrier elements of barrier element array 1610 to be transparent or opaque. In this manner, barrier element array 1610 filters light 1652 to generate filtered light 1672 that includes one or more two-dimensional and/or three-dimensional images that may be viewed by users 1618 in viewing space 1670. Example further description of implementations of the

display driver circuits described herein is provided in pending U.S. patent application Ser. No. _____, titled “Integrated Backlighting, Sub-Pixel and Display Driver Circuitry Supporting Adaptive 2D, Stereoscopic 3D and Multi-View 3D Displays,” filed on same date herewith, which is incorporated by reference herein in its entirety, although the driver circuits described herein are not limited to such implementations.

[0172] For example, drive signal 1614 may control sets of pixels of pixel array 1608 to each emit light representative of a respective image, to provide a plurality of images. Drive signal 1616 may control barrier elements of barrier element array 1610 to filter the light received from pixel array 1608 according to the provided images such that one or more of the images are received by users 1618 in two-dimensional form. For instance, drive signal 1616 may select one or more sets of barrier elements of barrier element array 1610 to be transparent, to transmit one or more corresponding two-dimensional images or views to users 1618. Furthermore, drive signal 1616 may control sections of barrier element array 1610 to include opaque and transparent barrier elements to filter the light received from pixel array 1608 so that one or more pairs of images or views provided by pixel array 1608 are each received by users 1618 as a corresponding three-dimensional image or view. For example, drive signal 1616 may select parallel strips of barrier elements of barrier element array 1610 to be transparent to form slits that enable three-dimensional images to be received by users 1618.

[0173] In embodiments, drive signal 1616 may be generated by barrier array driver circuit 1606 to configure one or more characteristics of barrier element array 1610. For example, drive signal 1616 may be generated to form any number of parallel strips of barrier elements of barrier element array 1610 to be transparent, to modify the number and/or spacing of parallel strips of barrier elements of barrier element array 1610 that are transparent, to select and/or modify an orientation of one or more strips of barrier elements of barrier element array 1610 that are transparent, to select one or more areas of barrier element array 1610 to include all transparent or all opaque barrier elements, etc.

[0174] FIG. 17 shows a block diagram of a display system 1700, which is another example of display device 1500 of FIG. 15, according to an embodiment. As shown in FIG. 17, system 1700 includes display device driver circuit 1602, a pixel array 1722, parallax barrier 1620, and a backlighting 1716. Parallax barrier 1620 includes barrier element array 1610 and backlighting 1716 includes a light element array 1736. Furthermore, display driver circuit 1602 includes a pixel array driver circuit 1728, barrier array driver circuit 1606, and a light source driver circuit 1730. These features of system 1700 are described as follows.

[0175] Backlighting 1716 is a backlight panel that emits light 1738. Light element array 1736 (or “backlight array”) of backlighting 1716 includes a two-dimensional array of light sources. Such light sources may be arranged, for example, in a rectangular grid. Each light source in light element array 1736 is individually addressable and controllable to select an amount of light emitted thereby. A single light source may comprise one or more light-emitting elements depending upon the implementation. In one embodiment, each light source in light element array 1736 comprises a single light-emitting diode (LED) although this example is not intended to

be limiting. Further description of implementations of backlighting 1716 and other backlighting implementations described herein is provided in pending U.S. patent application Ser. No. _____, titled "Backlighting Array Supporting Adaptable Parallax Barrier," filed on same date herewith, which is incorporated by reference herein in its entirety.

[0176] Parallax barrier 1620 is positioned proximate to a surface of backlighting 1716 (e.g., a surface of the backlight panel). As described above, barrier element array 1610 is a layer of parallax barrier 1620 that includes a plurality of barrier elements or blocking regions arranged in an array. Each barrier element of the array is configured to be selectively opaque or transparent. Barrier element array 1610 filters light 1738 received from backlighting 1716 to generate filtered light 1740. Filtered light 1740 is configured to enable a two-dimensional image or a three-dimensional image (e.g., formed by a pair of two-dimensional images in filtered light 1672) to be formed based on images subsequently imposed on filtered light 1740 by pixel array 1722.

[0177] Similarly to pixel array 1608 of FIG. 16, pixel array 1722 of FIG. 17 includes a two-dimensional array of pixels (e.g., arranged in a grid or other distribution). However, pixel array 1722 is not self-illuminating, and instead is a light filter that imposes images (e.g., in the form of color, grayscale, etc.) on filtered light 1740 from parallax barrier 1620 to generate filtered light 1672 to include one or more images. Each pixel of pixel array 1722 may be a separately addressable filter (e.g., a pixel of a plasma display, an LCD display, an LED display, or of other type of display). Each pixel of pixel array 1722 may be individually controllable to vary the color imposed on the corresponding light passing through, and/or to vary the intensity of the passed light in filtered light 1672. In an embodiment, each pixel of pixel array 1722 may include a plurality of sub-pixels that correspond to separate color channels, such as a trio of red, green, and blue sub-pixels included in each pixel.

[0178] Display driver circuit 1602 of FIG. 17 is configured to generate drive signals based on control signal 1622 and/or content signal 1624 to enable display system 1700 to display two-dimensional and three-dimensional images to users 1618 in viewing space 1670. For example, light source driver circuit 1730 within display driver circuit 1602 controls the amount of light emitted by each light source in light element array 1736 by generating a drive signal 1734 that is received by light element array 1736 (based on content signal 1624 and/or control signal 1622). Drive signal 1734 may include one or more drive signals used to control the amount of light emitted by each light source in light element array 1736 to generate light 1738. As described above, barrier array driver circuit 1606 is configured to generate drive signal 1616 received by barrier element array 1610 (e.g., based on control signal 1622). Drive signal 1616 may include one or more drive signals used to cause each of the barrier elements of barrier element array to be transparent or opaque, to filter light 1738 to generate filtered light 1740. Pixel array driver circuit 1728 is configured to generate a drive signal 1732 that is received by pixel array 1722 (e.g., based on content signal 1624 and/or control signal 1622). Drive signal 1732 may include one or more drive signals used to cause pixels of pixel array 1722 to impose desired images (e.g., colors, grayscale, etc.) on filtered light 1740 as it passes through pixel array 1722. In this manner, pixel array 1722 generates filtered light

1672 that includes one or more two-dimensional and/or three-dimensional images that may be viewed by users 1618 in viewing space 1670.

[0179] For example, drive signal 1734 may control sets of light sources of light element array 1736 to emit light 1738. Drive signal 1616 may control barrier elements of barrier element array 1610 to filter light 1738 received from light element array 1736 to enable filtered light 1740 to enable two- and/or three-dimensionality. Drive signal 1732 may control sets of pixels of pixel array 1722 to filter filtered light 1740 according to respective images, to provide a plurality of images. For instance, drive signal 1616 may select one or more sets of the barrier elements of barrier element array 1610 to be transparent, to enable one or more corresponding two-dimensional images to be delivered to users 1618. Furthermore, drive signal 1616 may control sections of barrier element array 1610 to include opaque and transparent barrier elements to filter the light received from light element array 1736 so that one or more pairs of images provided by pixel array 1722 are each enabled to be received by users 1618 as a corresponding three-dimensional image. For example, drive signal 1616 may select parallel strips of barrier elements of barrier element array 1610 to be transparent to form slits that enable three-dimensional images to be received by users 1618.

[0180] FIG. 18 shows a flowchart 1800 for generating images that are delivered to users in a viewing space, according to an exemplary embodiment. Flowchart 1800 may be performed by system 1600 in FIG. 16 or system 1700 of FIG. 17, for example. Flowchart 1800 is described with respect to FIG. 19, which shows a cross-sectional view of a display system 1900. Display system 1900 is an exemplary embodiment of system 1600 shown in FIG. 16, and is shown for purposes of illustration. As shown in FIG. 19, system 1900 includes a pixel array 1902 and a barrier element array 1904. In another embodiment, system 1900 may further include backlighting in a configuration similar to display system 1700 of FIG. 17. Further structural and operational embodiments will be apparent to persons skilled in the relevant art(s) based on the discussion regarding flowchart 1800. Flowchart 1800 is described as follows.

[0181] Flowchart 1800 begins with step 1802. In step 1802, light is received at an array of barrier elements. For example, as shown in FIG. 16, light 1652 is received at parallax barrier 1620 from pixel array 1608. Each pixel of pixel array 1608 may generate light that is received at parallax barrier 1620. Depending on the particular display mode of parallax barrier 1620, parallax barrier 1620 may filter light 1652 from pixel array 1608 to generate a two-dimensional image or a three-dimensional image viewable in viewing space 1670 by users 1618. As described above with respect to FIG. 17, alternatively, light 1738 may be received by parallax barrier 1620 from light element array 1736.

[0182] In step 1804, a first set of the barrier elements of the array of barrier elements is configured in the blocking state and a second set of the barrier elements of the array of barrier elements is configured in the non-blocking state to enable a viewer to be delivered a three-dimensional view. Three-dimensional image content may be provided for viewing in viewing space 1670. In such case, referring to FIG. 16 or 17, barrier array driver circuit 1606 may generate drive signal 1616 to configure barrier element array 1610 to include transparent strips of barrier elements to enable a three-dimensional view to be formed. For example, as shown in FIG. 19, barrier

element array **1904** includes a plurality of barrier elements that are each either transparent (in a non-blocking state) or opaque (in a blocking state). Barrier elements that are blocking are indicated as barrier elements **1910a-1910f**, and barrier elements that are non-blocking are indicated as barrier elements **1912a-1912e**. Further barrier elements may be included in barrier element array **1904** that are not visible in FIG. 19. Each of barrier elements **1910a-1910f** and **1912a-1912e** may include one or more barrier elements. Barrier elements **1910** alternate with barrier elements **1912** in series in the order of barrier elements **1910a**, **1912a**, **1910b**, **1912b**, **1910c**, **1912c**, **1910d**, **1912d**, **1910e**, **1912e**, and **1910f**. In this manner, blocking barrier elements **1910** are alternated with non-blocking barrier elements **1912** to form a plurality of parallel non-blocking or transparent slits in barrier element array **1904**.

[0183] For instance, FIG. 20 shows a view of a parallax barrier **2000** with transparent slits, according to an exemplary embodiment. Parallax barrier **2000** is an example of parallax barrier **1620** of FIGS. 16 and 17. As shown in FIG. 20, parallax barrier **2000** includes barrier element array **2002**, which includes a plurality of barrier elements **2004** arranged in a two-dimensional array. Furthermore, as shown in FIG. 20, barrier element array **2002** includes a plurality of parallel strips of barrier elements **2004** that are selected to be non-blocking to form a plurality of parallel non-blocking strips (or “slits”) **2006a-2006g**. As shown in FIG. 20, parallel non-blocking strips **2006a-2006g** (non-blocking slits) are alternated with parallel blocking or blocking strips **2008a-2008g** of barrier elements **2004** that are selected to be blocking. In the example of FIG. 20, non-blocking strips **2006a-2006g** and blocking strips **2008a-2008g** each have a width (along the x-dimension) of two barrier elements **2004**, and have lengths that extend along the entire y-dimension (twenty barrier elements **2004**) of barrier element array **2002**, although in other embodiments, may have alternative dimensions. Non-blocking strips **2006a-2006g** and blocking strips **2008a-2008g** form a parallax barrier configuration for parallax barrier **300**. The spacing (and number) of parallel non-blocking strips **2006** in barrier element array **2002** may be selectable by choosing any number and combination of particular strips of barrier elements **2004** in barrier element array **2002** to be non-blocking, to be alternated with blocking strips **2008**, as desired. For example, hundreds, thousands, or even larger numbers of non-blocking strips **2006** and blocking strips **2008** may be present in parallax barrier **300**.

[0184] FIG. 21 shows a parallax barrier **2100** that is another example of parallax barrier **1620** with parallel transparent slits, according to an embodiment. Similarly to parallax barrier **2000** of FIG. 20, parallax barrier **2100** has includes a barrier element array **2112**, which includes a plurality of barrier elements **2114** arranged in a two-dimensional array (28 by 1 array). Barrier elements **2114** have widths (along the x-dimension) similar to the widths of barrier elements **2004** in FIG. 20, but have lengths that extend along the entire vertical length (y-dimension) of barrier element array **2114**. As shown in FIG. 21, barrier element array **2112** includes parallel non-blocking strips **2006a-2006g** alternated with parallel blocking strips **2008a-2008g**. In the example of FIG. 21, parallel non-blocking strips **2006a-2006g** and parallel blocking strips **2008a-2008g** each have a width (along the x-dimension) of two barrier elements **2114**, and have lengths that extend along the entire y-dimension (one barrier element **314**) of barrier element array **2112**.

[0185] Referring back to FIG. 18, in step **1806**, the light is filtered at the array of barrier elements to form the three-dimensional view in a viewing space. Barrier element array **1610** of parallax barrier **1620** is configured to filter light **1652** received from pixel array **1608** (FIG. 16) or light **1738** received from light element array **1736** (FIG. 17) according to whether barrier element array **1610** is transparent or non-blocking (e.g., in a two-dimensional mode) or includes parallel non-blocking strips (e.g., in a three-dimensional mode). If one or more regions of barrier element array **1610** are transparent, those regions of barrier element array **1610** function as “all pass” filters to substantially pass all of light **1652** as filtered light **1672** to deliver one or more corresponding two-dimensional images generated by pixel array **1608** to viewing space **1670**, to be viewable as a two-dimensional images in a similar fashion as a conventional display. If barrier element array **1610** includes one or more regions having parallel non-blocking strips (e.g., as shown for barrier element array **2002** in FIGS. 20 and 21), those regions of barrier element array **1610** pass a portion of light **1652** as filtered light **1672** to deliver one or more corresponding three-dimensional images to viewing space **1670**.

[0186] For example, as shown in FIG. 19, pixel array **1902** includes a plurality of pixels **1914a-1914d** and **1916a-1916d**. Pixels **1914** alternate with pixels **1916**, such that pixels **1914a-1914d** and **1916a-1916d** are arranged in series in the order of pixels **1914a**, **1916a**, **1914b**, **1916b**, **1914c**, **1916c**, **1914d**, and **1916d**. Further pixels may be included in pixel array **1902** that are not visible in FIG. 19, including further pixels along the width dimension of pixel array **1902** (e.g., in the left-right directions) as well as pixels along a length dimension of pixel array **1902** (not visible in FIG. 19). Each of pixels **1914a-1914d** and **1916a-1916d** generates light, which emanates from display surface **1924** of pixel array **1902** (e.g., generally upward in FIG. 19) towards barrier element array **1904**. Some example indications of light emanating from pixels **1914a-1914d** and **1916a-1916d** are shown in FIG. 19 (as dotted lines), including light **1924a** and light **1918a** emanating from pixel **1914a**, light **1924b**, light **1918b**, and light **1924c** emanating from pixel **1914b**, etc.

[0187] Furthermore, light emanating from pixel array **1902** is filtered by barrier element array **1904** to form a plurality of images in a viewing space **1926**, including a first image **1906a** at a first location **1908a** and a second image **1906b** at a second location **1908b**. A portion of the light emanating from pixel array **1902** is blocked by blocking barrier elements **1910**, while another portion of the light emanating from pixel array **1902** passes through non-blocking barrier elements **1912**, according to the filtering by barrier element array **1904**. For instance, light **1924a** from pixel **1914a** is blocked by blocking barrier element **1910a**, and light **1924b** and light **1924c** from pixel **1914b** are blocked by blocking barrier elements **1910b** and **1910c**, respectively. In contrast, light **1918a** from pixel **1914a** is passed by non-blocking barrier element **1912a** and light **1918b** from pixel **1914b** is passed by non-blocking barrier element **1912b**.

[0188] By forming parallel non-blocking slits in a barrier element array, light from a pixel array can be filtered to form multiple images or views in a viewing space. For instance, system **1900** shown in FIG. 19 is configured to form first and second images **1906a** and **1906b** at locations **1908a** and **1908b**, respectively, which are positioned at a distance **1928** from pixel array **1902** (as shown in FIG. 19, further instances of first and second images **1906a** and **1906b** may be formed in

viewing space **1926** according to system **1900**, in a repeating, alternating fashion). As described above, pixel array **1902** includes a first set of pixels **1914a-1914d** and a second set of pixels **1916a-1916d**. Pixels **1914a-1914d** correspond to first image **1906a** and pixels **1916a-1916d** correspond to second image **1906b**. Due to the spacing of pixels **1914a-1914d** and **1916a-1916d** in pixel array **1902**, and the geometry of non-blocking barrier elements **1912** in barrier element array **1904**, first and second images **1906a** and **1906b** are formed at locations **1908a** and **1908b**, respectively. As shown in FIG. **19**, light **1918a-1918d** from the first set of pixels **1914a-1914d** is focused at location **1908a** to form first image **1906a** at location **1908a**. Light **1920a-1920d** from the second set of pixels **1916a-1916d** is focused at location **1908b** to form second image **1906b** at location **1908b**.

[0189] FIG. **19** shows a slit spacing **1922** (center-to-center) of non-blocking barrier elements **1912** in barrier element array **1904**. Spacing **1922** may be determined to select locations for parallel non-blocking slits to be formed in barrier element array **1904** for a particular image distance **1928** at which images are desired to be formed (for viewing by users). For example, in an embodiment, if a spacing of pixels **1914a-1914d** corresponding to an image is known, and a distance **1928** at which the image is desired to be displayed is known, the spacing **1922** between adjacent parallel non-blocking slits in barrier element array **1904** may be selected.

[0190] First and second images **1906a** and **1906b** are configured to be perceived by a user as a three-dimensional image or view. For example, a viewer may receive first image **1906a** at a first eye location and second image **1906b** at a second eye location, according to an exemplary embodiment. First and second images **1906a** and **1906b** may be generated by first set of pixels **1914a-1914d** and second set of pixels **1916a-1916d** as images that are slightly different perspective from each other. Images **1906a** and **1906b** are combined in the visual center of the brain of the viewer to be perceived as a three-dimensional image or view. In such an embodiment, first and second images **1906a** and **1906b** may be formed by display system **1900** such that their centers are spaced apart a width of a user's pupils (e.g., an "interocular distance").

[0191] Note that in the embodiments of FIGS. **20** and **21**, the entire regions of parallax barriers **2000** and **2100** are filled with parallel non-blocking strips (e.g., as shown for barrier element array **2002** in FIGS. **20** and **21**) to be configured to deliver three-dimensional images to viewing space **1670**. In further embodiments, one or more regions of a parallax barrier may be filled with parallel non-blocking strips to deliver three-dimensional images, and one or more other regions of the parallax barrier may be transparent to deliver two-dimensional images. Furthermore, different regions of a parallax barrier that have parallel non-blocking strips may have the parallel non-blocking strips oriented at different angles to deliver three-dimensional images to viewers that are oriented differently.

[0192] For instance, FIG. **22** shows a view of a parallax barrier **2200** configured to enable the simultaneous display of two-dimensional and three-dimensional images at different regions, according to exemplary embodiments. Parallax barrier **2200** is similar to parallax barrier **2000** of FIG. **20**, having barrier element array **2002** including a plurality of barrier elements **2004** arranged in a two-dimensional array. In FIG. **22**, a first region **2202** of barrier element array **2002** includes a plurality of parallel non-blocking strips alternated with parallel blocking strips that together fill first region **2202**. A

second region **2204** of barrier element array **2002** is surrounded by first region **2202**. Second region **2204** is a rectangular shaped region of barrier element array **2002** that includes a two-dimensional array of barrier elements **2004** that are non-blocking. Thus, in FIG. **22**, barrier element array **2002** is configured to enable a three-dimensional image to be generated by pixels of a pixel array that are adjacent to barrier elements of first region **2202**, and to enable a two-dimensional image to be generated by pixels of the pixel array that are adjacent to barrier elements inside of second region **2204**. Note that alternatively, first region **2202** may include all non-blocking barrier elements **2002** to pass a two-dimensional image, and second region **2204** may include parallel non-blocking strips alternated with parallel blocking strips to pass a three-dimensional image. In further embodiments, parallax barrier **2200** may have additional numbers, sizes, and arrangements of regions configured to pass different combinations of two-dimensional images and three-dimensional images.

[0193] In another example, FIG. **23** shows a view of a parallax barrier **2300** with transparent slits having different orientations, according to an exemplary embodiment. Parallax barrier **2300** is similar to parallax barrier **2000** of FIG. **20**, having barrier element array **2002** including a plurality of barrier elements **2004** arranged in a two-dimensional array. A first region **2310** (e.g., a bottom half) of barrier element array **2002** includes a first plurality of parallel strips of barrier elements **2004** that are selected to be non-blocking to form a first plurality of parallel non-blocking strips **2302a-2302e** (each having a width of two barrier elements **2004**). As shown in FIG. **23**, parallel non-blocking strips **2302a-2302e** are alternated with parallel blocking strips **2304a-2304f** of barrier elements **2004** (each having a width of three barrier elements **2004**). Parallel non-blocking strips **2302a-2302e** are oriented in a first direction (e.g., along a vertical axis).

[0194] Furthermore, as shown in FIG. **23**, a second region **2312** (e.g., a top half) of barrier element array **2002** includes a second plurality of parallel strips of barrier elements **2004** that are selected to be non-blocking to form a second plurality of parallel non-blocking strips **2306a-2306d** (each having a width of one barrier element **2004**). As shown in FIG. **23**, parallel non-blocking strips **2306a-2306d** are alternated with parallel blocking strips **2308a-2308c** of barrier elements **2004** (each having a width of two barrier elements **2004**). Parallel non-blocking strips **2306a-2306d** are oriented in a second direction (e.g., along a horizontal axis).

[0195] As such, in FIG. **23**, first and second pluralities of parallel non-blocking strips **2302a-2302e** and **2306a-2306d** are present in barrier element array **2002** that are oriented perpendicularly to each other. The region of barrier element array **2002** that includes first plurality of parallel non-blocking strips **2302a-2302e** may be configured to deliver a three-dimensional image in a viewing space (as described above) to be viewable by a user whose body is oriented vertically (e.g., sitting upright or standing up). The region of barrier element array **2002** that includes second plurality of parallel non-blocking strips **2306a-2306d** may be configured to deliver a three-dimensional image in a viewing space (as described above) to be viewable by a user whose body is oriented horizontally (e.g., laying down). In this manner, users who are oriented differently relative to each other can still each be provided with a corresponding three-dimensional image that accommodates their position.

[0196] As described above, in an embodiment, display device 1502 of FIG. 15 may be configured to generate a two-dimensional image for viewing by users in a viewing space. For instance, referring to FIGS. 16 and 17, barrier element array 1610 may be configured into a third configuration to deliver a two-dimensional view. In the third configuration, barrier array driver circuit 1606 may generate drive signal 1616 to configure each barrier element of barrier element array 1610 to be in the non-blocking state (transparent). If barrier element array 1610 is non-blocking, barrier element array 1610 functions as an “all pass” filter to substantially pass all of light 1652 (FIG. 16) or light 1738 (FIG. 17) as filtered light 1672 to deliver the two-dimensional image to viewing space 1670, to be viewable as a two-dimensional image in a similar fashion as a conventional display.

[0197] In embodiments, display systems may be configured to generate multiple two-dimensional images or views for viewing by users in a viewing space. For example, FIG. 24 shows a display system 2400 configured to deliver two two-dimensional images, according to an embodiment. Display system 2400 is configured similarly to display system 1900 of FIG. 19. As shown in FIG. 24, display system 2400 includes pixel array 1902 and barrier element array 1904, which generate first and second images 2402a and 2402b. As shown in FIG. 24, a first viewer 2404a receives first image 2402a at a first location and a second viewer 2404b receives second image 2402b at a second location, according to an exemplary embodiment. Similarly to the description provided above with respect to FIG. 19, first and second images 2402a and 2402b may be generated by first set of pixels 1914a-1914d and second set of pixels 1916a-1916d of pixel array 1902. However, rather than first and second images 2402a and 2402b being images that are of different perspective, first and second images 2402a and 2402b are each a two-dimensional image that may be viewed independently from each other. For instance, image 2402a and image 2402b may be generated by display system 1900 from first media content and second media content, respectively, that are independent of each other. Image 2402a may be received by both eyes of first viewer 2404a to be perceived by first viewer 2404a as a first two-dimensional image, and image 2402b may be received by both eyes of second viewer 2404b to be perceived by second viewer 2404b as a second two-dimensional image. Thus, first and second images 2402a and 2402b may be generated to have a spacing that enables them to be separately viewed by first and second users 2404a and 2404b.

[0198] As such, display system 2400 of FIG. 24 can be configured to deliver a single three-dimensional view to a viewer (e.g., as shown in FIG. 19 for display system 1900), to deliver a pair of two-dimensional views to a pair of viewers (e.g., as shown in FIG. 24), or to deliver a pair of three-dimensional views to a pair of viewers as described above). Display system 2400 can be configured to switch between delivering views to one and two viewers by turning off or turning on, respectively, the display of media content by pixel array 1902 associated with one of the viewers (e.g., by turning off or on pixels 1916 associated with second image 2402b). Display system 2400 can be configured to switch between delivering two-dimensional and three-dimensional views by providing the corresponding media content type at pixel array 2402. Furthermore, display system 2400 may provide such capabilities when configured similarly to display system 1700 shown in FIG. 17 (e.g., including backlighting 1716).

[0199] In an embodiment, display system 1900 may be configured to generate multiple three-dimensional images that include related image content (e.g., each three-dimensional image is a different viewpoint of a common scene), or that each include unrelated image content, for viewing by users in a viewing space. Each of the three-dimensional images may correspond to a pair of images generated by pixels of the pixel array. The barrier element array filters light from the pixel array to form the image pairs in a viewing space to be perceived by users as three-dimensional images.

[0200] For instance, FIG. 25 shows a flowchart 2500 for generating multiple three-dimensional images, according to an exemplary embodiment. Flowchart 2500 is described with respect to FIG. 26, which shows a cross-sectional view of a display system 2600. Display system 2600 is an exemplary embodiment of system 1600 shown in FIG. 16, and is shown for purposes of illustration. As shown in FIG. 26, system 2600 includes a pixel array 2602 and a barrier element array 2604. System 2600 may also include display driver circuit 1602 of FIG. 16, which is not shown in FIG. 26 for ease of illustration. Further structural and operational embodiments will be apparent to persons skilled in the relevant art(s) based on the discussion regarding flowchart 2500. Flowchart 2500 is described as follows.

[0201] Flowchart 2500 begins with step 2502. In step 2502, light is received from an array of pixels that includes a plurality of pairs of sets of pixels. For instance, in the example of FIG. 26, pixel array 2602 includes a first set of pixels 2614a-2614d, a second set of pixels 2616a-2616d, a third set of pixels 2618a-2618d, and a fourth set of pixels 2620a-2620d. Each of pixels 2614a-2614d, 2616a-2616d, 2618a-2618d, 2620a-2620d generates light, which emanates from the surface of pixel array 2602 towards barrier element array 2604. Each set of pixels generates a corresponding image. First set of pixels 2614a-2614d and third set of pixels 2618a-2618d are configured to generate images that combine to form a first three-dimensional image. Second set of pixels 2616a-2616d and fourth set of pixels 2620a-2620d are configured to generate images that combine to form a second three-dimensional image. Pixels of the four sets of pixels are alternated in pixel array 2602 in the order of pixel 2614a, pixel 2616a, pixel 2618a, pixel 2620a, pixel 2614b, pixel 2616b, etc. Further pixels may be included in each set of pixels in pixel array 2602 that are not visible in FIG. 26, including hundreds, thousands, or millions of pixels in each set of pixels.

[0202] As described above, in the current embodiment, pixel array 2602 is segmented into a plurality of pairs of sets of pixels. For instance, in the example of FIG. 26, pixel array 2602 is segmented into four sets of pixels. The first set of pixels includes pixels 2614a-2614g and the other pixels in the same columns, the second set of pixels includes pixels 2616a-2616g and the other pixels in the same columns, pixels 2618a-2618g and the other pixels in the same columns, and pixels 2620a-2620g and the other pixels in the same columns.

[0203] In step 2504, a plurality of strips of barrier elements of a barrier element array is selected to be non-blocking to form a plurality of parallel non-blocking slits. As shown in FIG. 26, barrier element array 2604 includes barrier elements that are each either non-blocking or blocking. Barrier elements that are blocking are indicated as barrier elements 2610a-2610f, and barrier elements that are non-blocking are indicated as barrier elements 2612a-2612e. Further barrier elements may be included in barrier element array 2604 that are not visible in FIG. 26, including hundreds, thousands, or

millions of barrier elements, etc. Each of barrier elements **2610a-2610f** and **2612a-2612e** may include one or more barrier elements. Barrier elements **2610** alternate with barrier elements **2612**. In this manner, blocking barrier elements **2610** are alternated with non-blocking barrier elements **2612** to form a plurality of parallel non-blocking slits in barrier element array **2604**.

[0204] In step **2506**, the light is filtered at the barrier element array to form a plurality of pairs of images in a viewing space corresponding to the plurality of pairs of sets of pixels, each pair of images of the plurality of pairs of images being configured to be perceived as a corresponding three-dimensional image of a plurality of three-dimensional images. As shown in FIG. **26**, light emanating from pixel array **2602** is filtered by barrier element array **2604** to form a plurality of images in a viewing space **2626**. For instance, four images are formed in viewing space **2626**, including first-fourth images **2606a-2606d**. Pixels **2614a-2614d** correspond to first image **2606a**, pixels **2616a-2616d** correspond to second image **2606b**, pixels **2618a-2618d** correspond to third image **2606c**, and pixels **2620a-2620d** correspond to fourth image **2606d**. As shown in FIG. **26**, light **2622a-2622d** from the first set of pixels **2614a-2614d** forms first image **2606a**, and light **2624a-2624d** from the third set of pixels **2618a-2618d** forms third image **2606c**, due to the filtering of the non-blocking slits (corresponding to non-blocking barrier elements **2612a-2612e**) in barrier element array **2604**. Although not indicated in FIG. **26** (for ease of illustration), in a similar fashion, light from the second set of pixels **2616a-2616d** forms second image **2606b**, and light from the fourth set of pixels **2620a-2620d** forms fourth image **2606d**.

[0205] In the embodiment of FIG. **26**, any pair of images of images **2606a-2606d** may be configured to be perceived as a three-dimensional image by a user in viewing space **2626**. For instance, first and third images **2606a** and **2606c** may be configured to be perceived by a user as a first three-dimensional image, such that first image **2606a** is received at a first eye location and third image **2606c** is received at a second eye location of a user. Furthermore, second and fourth images **2606b** and **2606d** may be configured to be perceived by a user as a second three-dimensional image, such that second image **2606b** is received at a first eye location and fourth image **2606d** is received at a second eye location of a user.

[0206] In the example of FIG. **26**, two three-dimensional images are provided by system **2600**. In further embodiments, further numbers of three-dimensional images may be provided, including a third three-dimensional image, a fourth three-dimensional image, etc. In such case, each three-dimensional image is generated by filtering light (using a barrier element array) corresponding to an image pair generated by a corresponding pair of sets of pixels of the pixel array, in a similar fashion as described with respect to FIG. **26** for two three-dimensional images. For example, to provide three three-dimensional images, pixel array **2602** may include fifth and sixth sets of pixels that generate fifth and sixth images, respectively, to be perceived by a user as a third three-dimensional image. To provide a fourth three-dimensional image, pixel array **2602** may include seventh and eighth sets of pixels that generate seventh and eighth images, respectively, to be perceived by a user as the fourth three-dimensional image.

[0207] In FIG. **26**, the first and second three-dimensional images generated based on first and third images **2606a** and **2606c** and second and fourth images **2606b** and **2606d**, respectively, and any further three-dimensional images that

may be generated, may include related image content or may each include unrelated image content. For example, in an embodiment, the first and second three-dimensional images (and any further three-dimensional images) may have been captured as different viewpoints of a common scene. Thus, a user in viewing space **2626** that moves laterally to sequentially view the first and second three-dimensional images (and any further three-dimensional images) may perceive being able to partially or fully “view behind” objects of the common scene.

[0208] Further description regarding using a parallax barrier to deliver three-dimensional views, including adaptable versions of parallax barriers, is provided in pending U.S. patent application Ser. No. 12/845,409, titled “Display With Adaptable Parallax Barrier,” in pending U.S. patent application Ser. No. 12/845,440, titled “Adaptable Parallax Barrier Supporting Mixed 2D And Stereoscopic 3D Display Regions,” and in pending U.S. patent application Ser. No. 12/845,461, titled “Display Supporting Multiple Simultaneous 3D Views,” which are each incorporated by reference herein in their entireties.

2. Exemplary Embodiments Using Lenticular Lenses

[0209] In embodiments, as described herein, display devices **1400** and **1500** of FIGS. **14** and **15** may include one or more lenticular lenses as light manipulators **1404** and **1504** used to deliver three-dimensional images and/or two-dimensional images. For instance, display systems **1600** and **1700** of FIGS. **16** and **17** may each include a sub-lens array of a lenticular lens in place of parallax barrier **1620**. For example, FIG. **27** shows a perspective view of a lenticular lens **2700** in accordance with an embodiment. As shown in FIG. **27**, lenticular lens **2700** includes a sub-lens array **2702**. Sub-lens array **2702** includes a plurality of sub-lenses **2704** arranged in a two-dimensional array (e.g., arranged side-by-side in a row). Each sub-lens **2704** is shown in FIG. **27** as generally cylindrical in shape and having a substantially semi-circular cross-section, but in other embodiments may have other shapes. In FIG. **27**, sub-lens array **2702** is shown to include eight sub-lenses for illustrative purposes and is not intended to be limiting. For instance, sub-lens array **2702** may include any number (e.g., hundreds, thousands, etc.) of sub-lenses **2704**. FIG. **28** shows a side view of lenticular lens **2700**, oriented as lenticular lens **2700** may be positioned in system **1900** of FIG. **19** (in place of parallax barrier **1904**) for lenticular lens **1902** to deliver three-dimensional views. In FIG. **28**, light may be passed through lenticular lens **2700** in the direction of dotted arrow **2802** to be diverted.

[0210] In one embodiment, lenticular lens **2700** may be fixed in size. For example, light manipulator **1404** of FIG. **14** may include lenticular lens **2700** when fixed in size. In another embodiment, lenticular lens **2700** may be adaptable. For instance, light manipulator **1504** of FIG. **15** may include lenticular lens **2700** when adaptable. For instance, in an embodiment lenticular lens **2700** may be made from an elastic material. Such a lenticular lens **2700** may be adapted in size in response to generated drive signals.

[0211] Further description regarding using a lenticular lens to deliver three-dimensional views, including adaptable versions of lenticular lenses, is provided in pending U.S. patent

application Ser. No. 12/774,307, titled “Display with Elastic Light Manipulator,” which is incorporated by reference herein in its entirety.

3. Exemplary Embodiments Using Multiple Light Manipulators

[0212] Display devices **1400** and **1500** may include multiple layers of light manipulators in embodiments. Multiple three-dimensional images may be displayed in a viewing space using multiple light manipulator layers, according to embodiments. In embodiments, the multiple light manipulating layers may enable spatial separation of the images. For instance, in such an embodiment, for example, a display device that includes multiple light manipulator layers may be configured to display a first three-dimensional image in a first region of a viewing space (e.g., a left-side area), a second three-dimensional image in a second region of the viewing space (e.g., a central area), a third three-dimensional image in a third region of the viewing space (e.g., a right-side area), etc. In embodiments, a display device may be configured to display any number of spatially separated three-dimensional images, as desired for a particular application (e.g., according to a number and spacing of viewers in the viewing space, etc.).

[0213] For instance, FIG. 29 shows a flowchart **2900** for generating multiple three-dimensional images using multiple light manipulator layers, according to an exemplary embodiment. Flowchart **2900** is described with respect to FIG. 30, which shows a cross-sectional view of a display system **3000** that includes multiple light manipulator layers, according to an exemplary embodiment. As shown in FIG. 30, system **3000** includes a display driver circuit **3002**, an image generator **1612**, a first light manipulator **3014a**, and a second light manipulator **3014b**. As shown in FIG. 30, image generator **1612** includes pixel array **1608**, first light manipulator **3014a** includes first light manipulator elements **3016a**, and second light manipulator **3014b** includes second light manipulator elements **3016b**. Furthermore, as shown in FIG. 30, display driver circuit **3002** includes a pixel array driver circuit **3004** and a light manipulator driver circuit **3006**. Flowchart **2900** and system **3000** are described as follows.

[0214] Flowchart **2900** begins with step **2902**. In step **2902**, light is received from an array of pixels that includes a plurality of pairs of sets of pixels. For example, as shown in FIG. 30, light **1652** is received at first light manipulator **3014a** from pixel array **208** of image generator **1612**. Pixel array driver circuit **3004** may generate driver signals based on content signal **1624** received by display driver circuit **3002**, and the driver signals may be received by pixel array **1614** to generate light **1652**. Each pixel of pixel array **1608** may generate light that is received at first light manipulator **3014a**. In an embodiment, pixel array driver circuit **3004** may generate drive signal **1614** to cause pixel array **1608** to emit light **1652** containing a plurality of images corresponding to the sets of pixels.

[0215] In step **2904**, the light from the array of pixels is manipulated with a first light manipulator. For example, first light manipulator **3014a** may be configured to manipulate light **1652** received from pixel array **1608**. As shown in FIG. 30, first light manipulator **3014a** includes light manipulator elements **3016a** configured to perform manipulating (e.g., filtering, diverting, etc.) of light **1652** to generate manipulated light **1672**. Light manipulator elements **3016a** may optionally be configurable to adjust the manipulating performed by first light manipulator **3014a**. First light manipulator **3014a** may

perform filtering in a similar manner as a parallax barrier described above or in other manner. In another embodiment, first light manipulator **3014a** may include a lenticular lens that diverts light **1652** to perform light manipulating, generating manipulated light **1672**. In an embodiment, light manipulator driver circuit **3006** may generate drive signal **1616a** based on control signal **1622** received by display driver **3002** to cause light manipulator elements **3016a** to manipulate light **1652** as desired.

[0216] In step **2906**, the light manipulated by the first light manipulator is manipulated with a second light manipulator to form a plurality of pairs of images corresponding to the plurality of pairs of sets of pixels in a viewing space. For example, as shown in FIG. 30, manipulated light **1672** is received by second light manipulator **3014b** to generate manipulated light **3008** that includes a plurality of three-dimensional images **3010a-3010n** formed in viewing space **1670**. As shown in FIG. 30, second light manipulator **3014b** includes light manipulator elements **3016b** configured to perform manipulating of manipulated light **1672** to generate manipulated light **3008**. Light manipulator elements **3016b** may optionally be configurable to adjust the manipulating performed by second light manipulator **3014b**. In an embodiment, light manipulator driver circuit **3006** may generate drive signal **1616b** based on control signal **1622** to cause light manipulator elements **3016b** to manipulate manipulated light **1652** to generate manipulated light **3008** including three-dimensional images **3010a-3010n** as desired. In embodiments, second light manipulator **3014b** may include a parallax barrier or a lenticular lens configured to manipulate manipulated light **1652** to generate manipulated light **3008**.

[0217] As such, display system **3000** has a single viewing plane or surface (e.g., a plane or surface of pixel array **1608**, first light manipulator **3014a**, second light manipulator **3014b**) that supports multiple viewers with media content in the form of three-dimensional images or views. The single viewing plane of display system **3000** may provide a first three-dimensional view based on first three-dimensional media content to a first viewer, a second three-dimensional view based on second three-dimensional media content to a second viewer, and optionally further three-dimensional views based on further three-dimensional media content to further viewers. First and second light manipulators **3014a** and **3014b** each cause three-dimensional media content to be presented to a corresponding viewer via a corresponding area of the single viewing plane, with each viewer being enabled to view corresponding media content without viewing media content directed to other viewers. Furthermore, the areas of the single viewing plane that provide the various three-dimensional views of media content overlap each other at least in part. In the embodiment of FIG. 30, the areas may be the same area—an area of a display screen or surface of display system **3000**. As such, multiple three-dimensional views that are each viewable by a corresponding viewer may be delivered by a single display viewing plane.

[0218] Display system **3000** may be configured in various ways to generate multiple three-dimensional images according to flowchart **2900**, in embodiments. Furthermore, as described below, embodiments of display system **3000** may be configured to generate two-dimensional views, as well as any combination of one or more two-dimensional views simultaneously with one or more three-dimensional views.

[0219] For instance, in an embodiment, delivery of three-dimensional images may be performed in system **3000** using

multiple parallax barriers. FIG. 31 shows a cross-sectional view of a display system 3100, according to an exemplary embodiment. Display system 3100 is an example of system 3000 shown in FIG. 30. As shown in FIG. 31, system 3100 includes a pixel array 3102, a first barrier element array 3104, and a second barrier element array 3106. System 3100 may also include display driver circuit 3002 of FIG. 30, which is not shown in FIG. 31 for ease of illustration. System 3100 is described as follows.

[0220] As shown in the example of FIG. 31, pixel array 3102 includes a first set of pixels 3114a-3114c, a second set of pixels 3116a-3116c, a third set of pixels 3118a-3118c, and a fourth set of pixels 3120a-3120c. Pixels of the four sets of pixels are alternated in pixel array 3102 in the order of pixel 3114a, pixel 3116a, pixel 3118a, pixel 3120a, pixel 3114b, pixel 3116b, etc. Further pixels may be included in each set of pixels in pixel array 3102 that are not visible in FIG. 31, including hundreds, thousands, or millions of pixels in each set of pixels.

[0221] Each of pixels 3114a-3114c, 3116a-3116c, 3118a-3118c, and 3120a-3120c is configured to generate light, which emanates from the surface of pixel array 3102 towards first barrier element array 3104. Each set of pixels is configured to generate a corresponding image. For example, FIG. 32 shows display system 3100, where pixels of pixel array 3102 emit light. Light from second set of pixels 3116a-3116c and first set of pixels 3114a-3114c is configured to generate third and fourth images 3206c and 3206d, respectively, which may be perceived together as a second three-dimensional image by a second viewer 2404b. Light from fourth set of pixels 3120a-3120c and third set of pixels 3118a-3118c is configured to generate first and second images 3206a and 3206b, respectively, which may be perceived together as a first three-dimensional image by a first viewer 2404a. The light emitted by the sets of pixels is filtered by first and second barrier element arrays 3104 and 3106 to generate the first and second three-dimensional images in respective desired regions of a user space 3202 adjacent to display system 3100.

[0222] First-fourth images 3206a-3206d may be formed in viewing space 3202 at a distance from pixel array 3102 and at a lateral location of viewing space 3202 as determined by a configuration of display system 3100 of FIG. 31, including a width and spacing of non-blocking slits in first barrier element array 3104, by a width and positioning of non-blocking slits in second barrier element array 3106, by a spacing between pixel array 3102 and first barrier element array 3104, and a spacing between first and second barrier element arrays 3104 and 3106.

[0223] In an embodiment, system 3000 of FIG. 30 may be configured similarly to display system 1700 of FIG. 17 to deliver three-dimensional images and/or two-dimensional images. For instance, in embodiments, system 3000 may include backlighting 1716 and pixel array 1722 separated by one or both of first and second light manipulators 3014a and 3014b. For example, FIG. 33 shows a block diagram of a display system 3300, which is an example of display devices 1400 and 1500 shown in FIGS. 14 and 15, according to an embodiment. Display system 3300 is configured to display multiple three-dimensional images in a viewing space in a spatially separated manner. As shown in FIG. 33, system 3300 includes display driver circuit 3002, backlighting 1716, first light manipulator 3014a, second light manipulator 3014b, and pixel array 1722. As shown in FIG. 33, backlighting 1716 optionally includes light element array 1736, first

light manipulator 3014a includes first light manipulator elements 3016a, and second light manipulator 3014b includes second light manipulator elements 3016b. Furthermore, as shown in FIG. 33, display driver circuit 3002 receives control signal 1622 and content signal 1624 and includes light source driver circuit 1730, light manipulator driver circuit 3006, and pixel array driver circuit 1728. Light source driver circuit 1730, light manipulator driver circuit 3006, and pixel array driver circuit 1728 may generate drive signals to perform their respective functions based on control signal 1622 and/or content signal 1624. As shown in FIG. 33, first and second light manipulators 3014a and 3014b are positioned between backlighting 1716 and pixel array 1722. In another embodiment, pixel array 1722 may instead be located between first and second light manipulators 3014a and 3014b.

[0224] As shown in FIGS. 16 and 17, display driver circuit 1602 receives content signal 1624, and as shown in FIGS. 30 and 33, display driver circuit 3002 receives content signal 1624. Content signal 1624 is an example of content signals 1408 and 1508 of FIGS. 14 and 15. Content signal 1624 includes two-dimensional and/or three-dimensional content for display by the respective display devices/systems. For instance, display driver circuits 1602 and 3002 generate respective drive signals (e.g., pixel array drive signals) based on content signal 1624 to enable the content carried by content signal 1624 to be displayed.

D. Example Display Environments

[0225] As described above, light manipulators may be reconfigured to change the locations of delivered views based on changing viewer positions. As such, a position of a viewer may be determined/tracked so that a parallax barrier and/or light manipulator may be reconfigured to deliver views consistent with the changing position of the viewer. For instance, with regard to a parallax barrier, a spacing, number, arrangement, and/or other characteristic of slits may be adapted according to the changing viewer position. With regard to a lenticular lens, a size of the lenticular lens may be adapted (e.g., stretched, compressed) according to the changing viewer position. In embodiments, a position of a viewer may be determined/tracked by determining a position of the viewer directly, or by determining a position of a device associated with the viewer (e.g., a device worn by the viewer, held by the viewer, sitting in the viewer's lap, in the viewer's pocket, sitting next the viewer, etc.).

[0226] Examples of display environments for display embodiments described herein include environments having a single viewer, as well as environments having multiple viewers. For example, in one type of environment (e.g., an office, living room, etc.), a single viewer interacts with an electronic device, mobile or stationary, to view and/or interact with mixed 2D and 3D content, such as a mobile or desktop computer, smart phone, television, or other mobile or stationary device. It is noted that this type of environment may include more than one viewer. In another type of environment (e.g., a living room, a home theatre room, etc.), multiple viewers are enabled to interact with an electronic device, such as a television set (e.g., high-def, small screen, large screen, etc.), to view and/or interact with mixed 2D and 3D content in the form of television content, movies, video games, etc.

[0227] For instance, FIG. 34 shows a block diagram of a display environment 3400, according to an exemplary embodiment. In the example of FIG. 34, first and second viewers 3406a and 3406b are present in display environment

3400, and are enabled to interact with a display device **3402** to be delivered two-dimensional and/or three-dimensional media content. Although two viewers **3406** are shown present in FIG. **34**, in other embodiments, other numbers of viewers **3406** may be present in display environment **3400** that may interact with display device **3402** and may be delivered media content by display device **3402**. As shown in FIG. **34**, display environment **3400** includes display device **3402**, a first remote control **3404a**, a second remote control **3404b**, a first headset **3412a**, a second headset **3412b**, and viewers **3406a** and **3406b**. Display device **3402** is an example of the display devices described above, and may be configured similarly to any display device described herein, including display device **606**. Viewer **3406a** is delivered a view **3408a** by display device **3402**, and viewer **3406b** is delivered a view **3408b** by display device **3402**. Views **3408a** and **3408b** may each be a two-dimensional view or a three-dimensional view. Furthermore, in embodiments, view **3408a** may be delivered to viewer **3406a**, but not be visible by viewer **3406b**, and view **3408b** may be delivered to viewer **3406b**, but not be visible by viewer **3406a**.

[0228] Remote control **3404a** is a device that viewer **3406a** may use to interact with display device **3402**, and remote control **3404b** is a device that viewer **3406b** may use to interact with display device **3402**. For example, as shown in FIG. **34**, viewer **3406a** may interact with a user interface of remote control **3404a** to generate a display control signal **3414a**, and viewer **3406b** may interact with a user interface of remote control **3404b** to generate a display control signal **3414b**. Display control signals **3414a** and **3414b** may be transmitted to display device **3402** using wireless or wired communication links. Display control signals **3414a** and **3414b** may be configured to select particular content desired to be viewed by viewers **3406a** and **3406b**, respectively. For example, display control signals **3414a** and **3414b** may select particular media content to be viewed (e.g., television channels, video games, DVD (digital video discs) content, video tape content, web content, etc.). Display control signals **3414a** and **3414b** may select whether such media content is desired to be viewed in two-dimensional or three-dimensional form by viewers **3406a** and **3406b**, respectively. Remote controls **3404a** and **3404b** may be television remote control devices, game controllers, smart phones, or other remote control type device.

[0229] Headsets **3412a** and **3412b** are worn by viewers **3406a** and **3406b**, respectively. Headsets **3412a** and **3412b** each include one or two speakers (e.g., earphones) that enable viewers **3406a** and **3406b** to hear audio associated with the media content of views **3408a** and **3408b**. Headsets **3412a** and **3412b** enable viewers **3406a** and **3406b** to hear audio of their respective media content without hearing audio associated the media content of the other of viewers **3406a** and **3406b**. Headsets **3412a** and **3412b** may each optionally include a microphone to enable viewers **3406a** and **3406b** to interact with display device **3402** using voice commands.

[0230] Display device **3402a**, headset **3412a**, and/or remote control **3404a** may operate to provide position information **3410a** regarding viewers **3406a** to display device **3402**, and display device **3402b**, headset **3412b**, and/or remote control **3404b** may operate to provide position information **3410b** regarding viewers **3406b** to display device **3402**. Display device **3402** may use position information **3410a** and **3410b** to reconfigure one or more light manipulators (e.g., parallax barriers and/or lenticular lenses) of display

device **3402** to enable views **3408a** and **3408b** to be delivered to viewers **3406a** and **3406b**, respectively, at various locations. For example, display device **3402a**, headset **3412a**, and/or remote control **3404a** may use positioning techniques to track the position of viewer **3406a**, and display device **3402b**, headset **3412b**, and/or remote control **3404b** may use positioning techniques to track the position of viewer **3406b**.

E. Example Electronic Device Implementations

[0231] Embodiments may be implemented in hardware, software, firmware, or any combination thereof. For example, browser **106**, mixed 2D/3D supporting logic **108**, API **302**, operating system **304**, display driver **306**, browser **400**, user interface **402**, rendering engine **404**, client application(s) **406**, networking module **408**, code interpreter **410**, web browser **490**, OS **432**, browser/rendering engine **442**, 2D/3Dx UI display **444**, networking module **446**, UI backend **448**, client(s) **450**, parser **452**, render tree preparation module **454**, rendered tree display **456**, 2D/3Dx support **458**, streaming server application **466**, user input interfaces **420**, 2D, 3Dx & mixed display driver interface **422**, shell operations **424**, 2D, 3Dx, mixed 2D and 3Dx, & mixed 3Dx and 3Dy translation services **426**, API supporting regional 2D/3Dx **428**, 2D only driver variant **434**, 3Dx only driver variant **436**, mixed 2D and 3Ds driver variant **438**, translation services **430a-430c**, display driver **604**, first translator **1102**, and/or second translator **1104** may be implemented as computer program code configured to be executed in one or more processors, and/or as circuit logic.

[0232] For instance, FIG. **35** shows a block diagram of an example implementation of an electronic device **3500**, according to an embodiment. In embodiments, electronic device **3500** may include one or more of the elements shown in FIG. **35**. As shown in the example of FIG. **35**, electronic device **3500** may include one or more processors (also called central processing units, or CPUs), such as a processor **3504**. Processor **3504** is connected to a communication infrastructure **3502**, such as a communication bus. In some embodiments, processor **3504** can simultaneously operate multiple computing threads.

[0233] Electronic device **3500** also includes a primary or main memory **3506**, such as random access memory (RAM). Main memory **3506** has stored therein control logic **3528A** (computer software), and data.

[0234] Electronic device **3500** also includes one or more secondary storage devices **3510**. Secondary storage devices **3510** include, for example, a hard disk drive **3512** and/or a removable storage device or drive **3514**, as well as other types of storage devices, such as memory cards and memory sticks. For instance, electronic device **3500** may include an industry standard interface, such a universal serial bus (USB) interface for interfacing with devices such as a memory stick. Removable storage drive **3514** represents a floppy disk drive, a magnetic tape drive, a compact disk drive, an optical storage device, tape backup, etc.

[0235] As shown in FIG. **35**, secondary storage devices **3510** may include an operating system **3532** and a browser **3534**. Embodiments for operating system **3532** (e.g., OS **304**, OS **432**, etc.) and for browser **3534** (e.g., browser **106**, browser **400**, browser **490**, etc.) are described in detail above.

[0236] Removable storage drive **3514** interacts with a removable storage unit **3516**. Removable storage unit **3516** includes a computer useable or readable storage medium **3524** having stored therein computer software **3528B** (con-

trol logic) and/or data. Removable storage unit **3516** represents a floppy disk, magnetic tape, compact disk, DVD, optical storage disk, or any other computer data storage device. Removable storage drive **3514** reads from and/or writes to removable storage unit **3516** in a well known manner.

[0237] Electronic device **3500** further includes a communication or network interface **3518**. Communication interface **3518** enables the electronic device **3500** to communicate with remote devices. For example, communication interface **3518** allows electronic device **3500** to communicate over communication networks or mediums **3542** (representing a form of a computer useable or readable medium), such as LANs, WANs, the Internet, etc. Network interface **3518** may interface with remote sites or networks via wired or wireless connections.

[0238] Control logic **3528C** may be transmitted to and from electronic device **3500** via the communication medium **3542**.

[0239] Any apparatus or manufacture comprising a computer useable or readable medium having control logic (software) stored therein is referred to herein as a computer program product or program storage device. This includes, but is not limited to, electronic device **3500**, main memory **3506**, secondary storage devices **3510**, and removable storage unit **3516**. Such computer program products, having control logic stored therein that, when executed by one or more data processing devices, cause such data processing devices to operate as described herein, represent embodiments of the invention.

[0240] Devices in which embodiments may be implemented may include storage, such as storage drives, memory devices, and further types of computer-readable media. Examples of such computer-readable storage media include a hard disk, a removable magnetic disk, a removable optical disk, flash memory cards, digital video disks, random access memories (RAMs), read only memories (ROM), and the like. As used herein, the terms “computer program medium” and “computer-readable medium” are used to generally refer to the hard disk associated with a hard disk drive, a removable magnetic disk, a removable optical disk (e.g., CDRoms, DVDs, etc.), zip disks, tapes, magnetic storage devices, MEMS (micro-electromechanical systems) storage, nano-technology-based storage devices, as well as other media such as flash memory cards, digital video discs, RAM devices, ROM devices, and the like. Such computer-readable storage media may store program modules that include computer program logic for browser **106**, mixed 2D/3D supporting logic **108**, API **302**, operating system **304**, display driver **306**, browser **400**, user interface **402**, rendering engine **404**, client application(s) **406**, networking module **408**, code interpreter **410**, web browser **490**, OS **432**, browser/rendering engine **442**, 2D/3Dx UI display **444**, networking module **446**, UI backend **448**, client(s) **450**, parser **452**, render tree preparation module **454**, rendered tree display **456**, 2D/3Dx support **458**, streaming server application **466**, user input interfaces **420**, 2D, 3Dx & mixed display driver interface **422**, shell operations **424**, 2D, 3Dx, mixed 2D and 3Dx, & mixed 3Dx and 3Dy translation services **426**, API supporting regional 2D/3Dx **428**, 2D only driver variant **434**, 3Dx only driver variant **436**, mixed 2D and 3Ds driver variant **438**, translation services **430a-430c**, display driver **604**, first translator **1102**, second translator **1104**, flowchart **500**, flowchart **700**, flowchart **1200** (including any one or more steps of flowcharts **500**, **700**, and **1200**), and/or further embodiments of the present invention described herein. Embodiments of

the invention are directed to computer program products comprising such logic (e.g., in the form of program code or software) stored on any computer useable medium (e.g., a computer readable storage medium). Such program code, when executed in one or more processors, causes a device to operate as described herein.

[0241] The invention can work with software, hardware, and/or browser implementations other than those described herein. Any software, hardware, and browser implementations suitable for performing the functions described herein can be used.

[0242] As described herein, electronic device **3500** may be implemented in association with a variety of types of display devices. For instance, electronic device **3500** may be one of a variety of types of media devices, such as a stand-alone display (e.g., a television display such as flat panel display, etc.), a computer, a game console, a set top box, a digital video recorder (DVR), other electronic device mentioned elsewhere herein, etc. Media content that is delivered in two-dimensional or three-dimensional form according to embodiments described herein may be stored locally or received from remote locations. For instance, such media content may be locally stored for playback (replay TV, DVR), may be stored in removable memory (e.g. DVDs, memory sticks, etc.), may be received on wireless and/or wired pathways through a network such as a home network, through Internet download streaming, through a cable network, a satellite network, and/or a fiber network, etc. For instance, FIG. **35** shows a first media content **3530A** that is stored in hard disk drive **3512**, a second media content **3530B** that is stored in storage medium **3524** of removable storage unit **3516**, and a third media content **3530C** that may be remotely stored and received over communication medium **3522** by communication interface **3518**. Media content **3530** may be stored and/or received in these manners and/or in other ways.

[0243] FIG. **36** shows a block diagram of a display system **3600** that supports mixed 2D, stereoscopic 3D and multi-view 3D displays according to an exemplary embodiment. Display system **3600** is another electronic device embodiment. As shown in FIG. **36**, display system **3600** includes media input interfaces **3602**, host processing circuitry **3604**, user input devices **3606**, display processing circuitry **3608**, adaptable display driver circuitry **3610**, adaptable 2D, 3Dx and mixed display **3612**, and first-third interface circuitry **3614-3618**. Host processing circuitry **3604** includes mixed 2D and 3Dx browser **490** (of FIG. **4B**), operating system **432** (of FIG. **4B**), and application programs **3622**. Display processing circuitry **3608** includes 2D, 3Dx, mixed 2D and 3Dx, and mixed 3Dx and 3Dy translation services **3640**.

[0244] Media input interfaces **3602** includes one or more media input interfaces, wired or wireless, for received media, such as those described elsewhere herein. For instance, media input interface **3602** may include an interface for receiving media content from a local media player device, such as a DVD player, a memory stick, a computer media player, etc., and may include commercially available (e.g., USB, HDMI, etc.) or proprietary interfaces for receiving local media content. Media input interface **3602** may include an interface for receiving media content from a remote source, such as the Internet, satellite, cable, etc.), and may include commercially available (e.g., WLAN, Data Over Cable Service Interface Specification (DOCSIS), etc.) or proprietary interfaces for receiving remote media content.

[0245] Host processing circuitry **3604** may include one or more integrated circuit chips and/or additional circuitry, which may be configured to execute software/firmware, including operating system **432**, browser **490**, and application programs **3622**.

[0246] User input devices **3606** includes one or more user input devices that a user may use to interact with display system **3600**. Examples of user input devices are described elsewhere herein, such as a keyboard, a mouse/pointer, etc.

[0247] Display processing circuitry **3608** may be included in host processing circuitry **3604**, or may be separate from host processing circuitry **3604** as shown in FIG. **36**. For instance, display processing circuitry **3608** may include one or more processors (e.g., graphics processors), further circuitry and/or other hardware, software, firmware, or any combination thereof. Display processing circuitry **3608** may be present to perform graphics processing tasks. For instance, as shown in FIG. **36**, display processing circuitry **3608** may optionally include 2D, 3Dx, mixed 2D and 3Dx, and mixed 3Dx and 3Dy translation services **3640** to perform 2D/3D related translation services in addition or alternatively to translation services of OS **432** and/or browser **490**.

[0248] Adaptable display driver circuitry **3610** includes one or more display driver circuits for an adaptable display. Examples of adaptable display driver circuitry **3610** are described above, such as with regard to FIGS. **4B**, **16**, **17**, **30**, and **33**.

[0249] Adaptable 2D, 3Dx and mixed display **3612** includes a display that is adaptable, and is capable of displaying 2D content, 3D content, and a mixture of 2D and/or 3D content. Examples of adaptable 2D, 3Dx and mixed display **3612** are described elsewhere herein.

[0250] First-third interface circuitry **3614-3618** is optional. For instance, as shown in FIG. **36**, a communication infrastructure (e.g., a signal bus) **3634** may be present to couple signals of media input interfaces **3602**, host processing circuitry **3604**, user input devices **3606**, display processing circuitry **3608**, adaptable display driver circuitry **3610**, and display **3612**. In an embodiment, if display processing circuitry **3608**, adaptable display driver circuitry **3610**, and/or display **3612** are contained in a common housing/structure with host processing circuitry **3604** (e.g., in a handheld device, etc.) interface circuitry **3614-3618** may not be needed to be present. If display processing circuitry **3608**, adaptable display driver circuitry **3610**, and/or display **3612** are in a separate housing/structure from host processing circuitry **3604**, corresponding interface circuitry **3614-3618** may be present to provide an interface. For instance, host processing circuitry **3604** may be in a game console, a desktop computer tower, a home audio receiver, a set top box, etc., and display processing circuitry **3608**, adaptable display driver circuitry **3610**, and/or display **3612** may be included in a display device structure. In such case, interface circuitry **3614-3618** may not be present. When present, first-third circuitry **3614-3618** may each include circuitry, such as receivers and/or transmitters (wired or wireless), for enabling communications between the respective one of display processing circuitry **3608**, adaptable display driver circuitry **3610**, and display **3612**, and the other components of system **3600** (e.g., host processing circuitry **3604**, etc.).

[0251] Note that the embodiment of display system **3600** shown in FIG. **36** is provided for purposes of illustration, and is not intended to be limiting. In further embodiments, display

system **3600** may include fewer, additional, and/or alternative features than shown in FIG. **36**.

IV. Conclusion

[0252] While various embodiments of the present invention have been described above, it should be understood that they have been presented by way of example only, and not limitation. It will be apparent to persons skilled in the relevant art that various changes in form and detail can be made therein without departing from the spirit and scope of the invention. Thus, the breadth and scope of the present invention should not be limited by any of the above-described exemplary embodiments, but should be defined only in accordance with the following claims and their equivalents.

What is claimed is:

1. A method used by a browser architecture to support display on a screen of two-dimensional content and three-dimensional content, the browser architecture receiving web page content, the method comprising:

- parsing the web page content;
- identifying two-dimensional content to be displayed in a first region of the screen;
- communicating a first configuration request to at least attempt to cause a first configuration of the first region of the screen to support the two-dimensional content;
- identifying three-dimensional content to be displayed in a second region of the screen; and
- communicating a second configuration request to at least attempt to cause a second configuration of the second region of the screen to support the three-dimensional content, the first configuration being different from the second configuration.

2. The method of claim 1, further comprising identifying at least part of the second configuration via interaction with a file containing the three-dimensional content.

3. The method of claim 1, further comprising identifying at least part of the second configuration via interaction with a server.

4. The method of claim 1, further comprising identifying at least part of the second configuration from the web page content.

5. The method of claim 4, wherein the web page content has a language format, and the identification of the at least part of the second configuration involves extracting configuration information via the language format.

6. A browser architecture that processes textually formatted content and supports display on a screen of two-dimensional content and three-dimensional content, the browser architecture comprising:

- a first browser portion that identifies a two-dimensional element and a three-dimensional element by parsing the textually formatted content;
- a second browser portion that identifies a screen configuration to be applied to a first region of the screen, the second browser portion delivering a configuration request based on the identification of the screen configuration to at least attempt to cause a reconfiguration of the screen within the first region; and
- a third browser portion that manages retrieval and display of the three-dimensional element within the second region of the screen.

7. The browser architecture of claim 6, wherein the first browser portion comprising an engine, and the third portion comprising a three dimensional presentation component.

8. The browser architecture of claim 7, wherein the three dimensional presentation component comprises a media player.

9. The browser architecture of claim 8, wherein the media player is an application that assists the engine.

10. The browser architecture of claim 6, wherein the first browser portion and the second browser portion both comprise an engine.

11. A method used by a browser architecture, the method comprising:

identifying first tag information associated with two-dimensional content, the two-dimensional content intended for both a left eye and a right eye of a viewer;

identifying second tag information associated with three-dimensional content, the three-dimensional content having a first portion and a second portion, the first portion intended for the left eye of the viewer and the second portion intended for the right eye of the viewer, the first portion being a first camera view and the second portion being a second camera view;

causing the presentation of the two-dimensional content in a first region of a screen; and

causing the presentation of the three-dimensional content in a second region of the screen.

12. The method of claim 11, wherein the causing the presentation of the three-dimensional content of the screen comprises delivering a configuration request to attempt to cause a reconfiguration of a region of the screen to support the three-dimensional content.

13. The method of claim 11, wherein the second tag information comprises an external link to three-dimensional content.

14. The method of claim 11, wherein the first tag information comprises textual information.

15. The method of claim 11, further comprising causing display of three-dimensional browser control elements.

16. The method of claim 11, wherein said causing the presentation of the three-dimensional content in a second region of the screen comprises:

responding to the second tag information by causing a control signal to be generated and received by a display driver, the control signal enabling the three-dimensional content to be displayed in the second region of the screen.

17. The method of claim 11, wherein said causing the presentation of the three-dimensional content in a second region of the screen comprises:

loading an information resource that includes the three-dimensional content, the information resource including at least one of a three-dimensional web page, a three-dimensional image, or a three-dimensional video.

18. The method of claim 11, wherein said causing the presentation of the three-dimensional content in a second region of the screen comprises:

executing a compiled script to generate the three-dimensional content to be displayed in the second region.

19. The method of claim 11, wherein the first region is a first tab of a browser window displayed in the screen, and the second region is a second tab of the browser window, the method further comprising:

displaying the two-dimensional content in the first tab of the browser window; and

displaying the three-dimensional content in the second tab of the browser window.

20. The method of claim 11, wherein the first region is a first frame of a browser window displayed in the screen, and the second region is a second frame of the browser window, the method further comprising:

displaying the two-dimensional content in the first frame of the browser window; and

displaying the three-dimensional content in the second frame of the browser window.

* * * * *