



(51) International Patent Classification:
G06F 16/78 (2019.01)

(21) International Application Number:
PCT/US2023/084864

(22) International Filing Date:
19 December 2023 (19.12.2023)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **GOOGLE LLC** [US/US]; 1600 Amphitheatre Parkway, Mountain View, California 94043 (US).

(72) Inventors: **JAIN, Palak**; GOOGLE LLC, No. 3, RMZ Infinity - Tower E Old Madras Road, Bangalore 560016 (IN). **YADAV, Vivek**; Google LLC, Bagmane Constellation Park, Bangalore 560337 (IN). **GUPTA, Shashank**; GOOGLE LLC, No. 3, RMZ Infinity - Tower E Old Madras Road, Bangalore 560016 (IN).

(74) Agent: **SCHOLZ, Jared** et al.; BRAKE HUGHES BELLERMANN LLP, P.O. Box 1077, Middletown, Maryland 21769 (US).

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

(84) Designated States (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ,

(54) Title: GENERATIVE AI BASED SKIP SCENE FEATURE

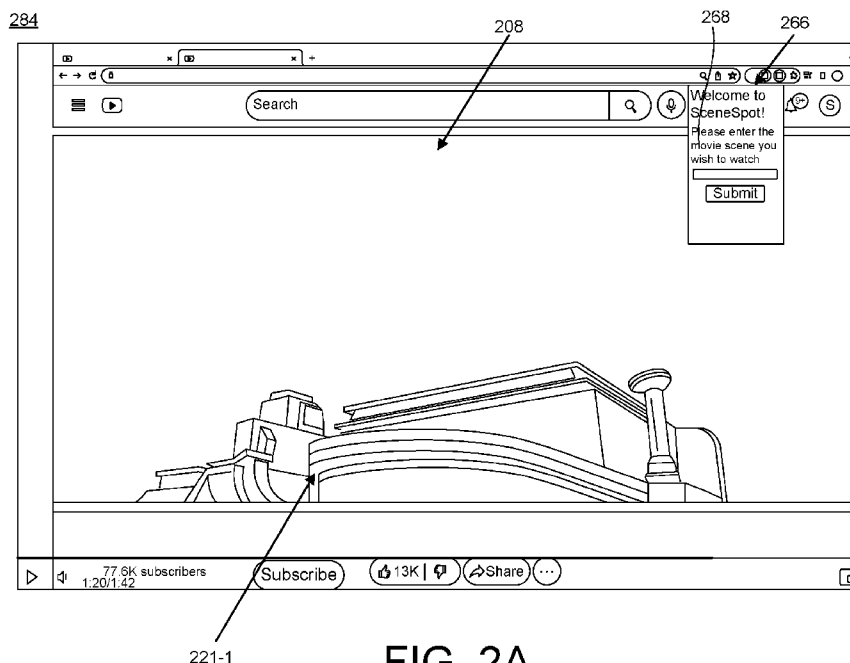


FIG. 2A

(57) Abstract: A device may initiate display of a media content item on a display device. A device may receive, via an interface of a media application, a natural language query about at least one scene in the media content item. A device may, in response to the natural language query, generate, by the media application, a prompt including a request to identify the at least one scene using the natural language query. A device may receive, from a machine-learning (ML) model, a prompt response that identifies at least one of a starting time or an ending time of the at least one scene. A device may programmatically navigate, by the media application, the media content item based on at least one of the starting time or the ending time of the at least one scene.

RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

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

GENERATIVE AI BASED SKIP SCENE FEATURE

BACKGROUND

[0001] Conventionally, a user may use an interface (UI) element (e.g., a forward or backward skip button, a timeline slider, etc.) to navigate to different scenes of a program provided by a streaming platform. In some examples, in order to navigate to a particular scene in the program, the user may move the timeline slider to view thumbnail images in various scenes to identify a particular scene of interest.

SUMMARY

[0002] In some aspects, the techniques described herein relate to a method including: initiating display of a media content item on a display device; receiving, via an interface of a media application, a natural language query about at least one scene in the media content item; in response to the natural language query, generating, by the media application, a prompt including a request to identify the at least one scene using the natural language query; receiving, from a machine-learning (ML) model, a prompt response that identifies at least one of a starting time or an ending time of the at least one scene; and programmatically navigating, by the media application, the media content item based on at least one of the starting time or the ending time of the at least one scene.

[0003] In some aspects, the techniques described herein relate to an apparatus including: at least one processor; and a non-transitory computer-readable medium storing executable instructions that cause the at least one processor to execute operations, the operations including: initiating display of a media content item on a display device; receiving, via an interface of a media application, a natural language query about at least one scene in the media content item; in response to the natural language query, generating, by the media application, a prompt including a request to identify the at least one scene using the natural language query; receiving, from a machine-learning (ML) model, a prompt response that identifies at least one of a starting time or an ending time of the at least one scene; and programmatically navigating, by the media application, the media content item based on at least one of the starting time or the ending time of the at least one scene.

[0004] In some aspects, the techniques described herein relate to a method including: generating, by a top scene generator, a prompt including a request to identify a plurality of top scenes in a media content item available for selection from a media platform; receiving,

from a machine-learning (ML) model, a prompt response that identifies a starting time and an ending time of each of the plurality of top scenes; and initiating display of a list of the plurality of top scenes for selection on a user interface of a display device; and in response to selection of a top scene from list, initiating playback of the media content item from the starting time.

[0005] The details of one or more implementations are set forth in the accompanying drawings and the description below. Other features will be apparent from the description and drawings, and from the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] FIG. 1 illustrates a streaming system that uses a machine-learning model to identify one or more scenes in a media content item based on a natural language query according to an aspect.

[0007] FIGS. 2A and 2B illustrate example user interfaces for navigating to a scene using a natural language query submitted via an interface according to an aspect.

[0008] FIGS. 3A and 3B illustrate example user interfaces for navigating to a scene using a natural language query submitted via an interface according to another aspect.

[0009] FIGS. 4A to 4C illustrate a streaming system according to another aspect.

[0010] FIG. 5 is a flowchart depicting example operations of identifying scenes in a media content item that is available for selection from a media platform using a machine-learning model according to an aspect.

DETAILED DESCRIPTION

[0011] This disclosure relates to a streaming system that includes a machine-learning (ML) model to identify a scene in a media content item based on a natural language query. The streaming system may also use the ML model to generate an edited version of a media content item by removing one or more scenes satisfying one or more content restriction categories. The system may increase the performance of a display device (e.g., a television device or other type of user device) by reducing latency and/or buffering issues caused by repeatedly navigating back and/or forth through a timeline of a media content item to locate a particular scene. For example, conventionally, a user may move a UI element (e.g., a skip forward button, a skip backward button, a timeline slider, etc.) to navigate to different parts of the media content item, but navigating back and forward in the item's timeline may cause

latency and/or buffering issues because repeatedly moving a current playback point may cause the display of content to be delayed due to network latency.

[0012] For instance, when moving to a new playback position, the display device starts to download the content and buffers it until there is enough content in the buffer to initiate playback. The resulting gap in playback in itself reduces the user experience, but significantly this delays the time when a user can start to assess whether the new playback position is a desired playback position. If it is not a desired playback position, the user will move a UI element (e.g., a skip forward button, a skip backward button, a timeline slider, etc.) to navigate to a further different part of the media content item, where buffering will cause a further gap in playback and a further delay to the time when a user can start to assess whether the new playback position is the desired playback position. This can occur multiple times in short succession. In addition to being inconvenient to the user, the buffering mechanism at the display device conventionally is complex in order to accommodate repeated changes in playback position for short periods of time, especially if content is to remain buffered in case of navigation to a position where the buffered content may again be played (e.g. in the case of skipping backwards in the media content). However, programmatically navigating to a particular scene using a natural language query may efficiently identify and render a scene of interest while avoiding latency and/or buffering issues otherwise caused by manually navigating back and/or forth through the item's timeline and may also avoid the need for complex buffering mechanisms.

[0013] A media application, executable by a display device, renders a user interface that identifies a plurality of media content items that are available for streaming on the display device. A media content item may be a show, program, movie, etc. Selection of a media content item from the application's user interface causes the media content item to be streamed on the display device. For example, in response to selection of a media content item from the user interface, the media application may initiate playback of the media content item. In some examples, during playback of the media content item, the media application may display an interface for receiving a natural language query about one or more scenes in the media content item. In response to the natural language query, the media application may generate a prompt with a request to identify one or more scenes in the media content item using the natural language query.

[0014] The media application may transmit the prompt to a ML model that is trained to identify scenes in media content items. In some examples, the ML model may be stored on a server computer. In some examples, the ML model may be stored on the display device.

In some examples, the display device is a virtual reality (VR) device or an augmented reality (AR) device, and the ML model is stored on a user device (e.g., the user's smartphone), which is connected to the VR device or the AR device. In response to the prompt, the ML model may identify the scene(s) that satisfy the natural language query. In some examples, the ML model includes a large language model (LLM). In some examples, the LLM is a conventional large language model (e.g., based on a transformer architecture), adapted to generate text in response to a text prompt provided as input. Such LLMs are trained on a large corpus of publicly available text, e.g., content from public databases and websites. In some examples, the LLM is configured to generate a textual response about the starting and/or ending times of one or more scenes that satisfy the natural language query. In some examples, the LLM is a specially trained language model (e.g., trained using media content available on one or more streaming platforms) that can identify scenes based on a natural language query.

[0015] The ML model generates and transmits a prompt response to the media application. The prompt response may include a starting time and/or an ending time of the scene(s) that satisfy the natural language query. In response to the prompt response, the media application may programmatically navigate the media content item based on the starting time and/or the ending time of the scene(s). In some examples, programmatically navigating includes navigating playback of the media content item to the starting time of the scene (e.g., skip to the identified scene). In some examples, programmatically navigating includes, during playback of the media content item, skipping the scene based on the starting time and the ending time of the scene. In some examples, the ML model identifies multiple scenes that correspond to the natural language query. For example, the prompt response may identify a list of scenes that satisfy the natural language query. The interface may include one or more navigation control elements that enable the user to skip to or go back to the scenes in the list of scenes that satisfy the natural language query.

[0016] In some examples, the media platform may programmatically generate edited versions (e.g., restricted versions) of media content items using the ML model. In some examples, the media platform may generate an edited version of a media content item by communicating with the ML model to identify one or more scenes that include restricted content and then removing the restricted content from playback of the media content item.

[0017] The media platform may store user accounts, where each user account stores information about a respective user. In some examples, a user account includes one or more profiles. A profile is associated with a particular user that is authorized under the user

account. A user account may include multiple profiles such as one or more adult profiles and one or more child profiles. The media platform may define one or more content restriction settings for a user account and/or a particular profile. In some examples, a profile may be associated with a content restriction setting (e.g., a parent control setting). In some examples, if the content restriction setting is enabled for a profile, the media platform may identify media content items under the profile that meet a certain rating (e.g., Motion Picture Association of America (MPAA) rating) (e.g., G rating, PG rating, etc.). In some examples, if a profile has an enabled content restriction setting, the media platform may provide an edited version of the media content item in the user interface. In some examples, the non-edited version may be a version that is not eligible to be viewed under the profile, but the edited version may be a version that is eligible to be viewed under the profile.

[0018] In some examples, the media platform may enable a user to customize an edited version of a media content item. For example, the media application may render a custom editing interface with a plurality of content restriction categories with respect to a particular media content item, a group of media content items, or all the media content items rendered under a profile or a user account. The content restriction categories may be different categories of content that may be deemed unsuitable for viewing such as violence, profanity, sexual content, drug use, and/or scary content. A user may use the custom editing interface to select one or more content restriction categories. In response to selection of the content restriction categories from the custom editing interface, the media application may generate an editing request, which causes generation of an edited version that corresponds to the user-selected content restriction categories to the media platform. For example, the ML model may receive identification of the media content item and the user-selected content restriction categories as inputs to the ML model, where the ML model identifies which scenes in the media content item include the user-selected content restriction categories. Using the starting and end times of the scenes identified by the ML model, the media platform may generate the edited version by removing the scenes that satisfy the user-selected content restriction categories.

[0019] In some examples, the media platform may communicate with a ML model to generate top scenes in a media content item that is available for selection in a user interface of a media application. For example, the media platform may include a top scene generator configured to generate a prompt for the ML model, where the prompt includes an instruction for the ML model to identify top scenes in a particular media content item. In some examples, the prompt includes information that identifies a particular media content item

(e.g., name of a program) with an instruction to identify the top scenes in the media content item. The ML model may identify the starting time and ending time of each top scene and transmit a prompt response with the starting and ending times of each top scene. In some examples, the prompt response also includes a short description of the top scene. In some examples, the ML model is a pre-trained large language model. The top scene generator may provide the list of top scenes (with their starting and ending times) to the media application, which renders the list of top scenes as selectable items on the user interface. In some examples, in response to selection of a top scene from the user interface, the media application causes the media content item to playback from the selected top scene. In some examples, the media content item continues to play from the end of the top scene. In some examples, after the selected top scene has ended, the media application may re-render the user interface that identifies the top scenes. In some examples, after the selected top scene has ended, the media application may render an interface that asks the user whether they want to continue with the media content item or return to the top scene selection interface.

[0020] In some examples, the media platform is configured to generate top scenes in real-time. For example, the top scenes in a particular media content item may change over time, and the media platform is configured to generate the top scenes in real-time (e.g., when the user requests a media content item's landing page). For example, in response to selection of a media content item among a plurality of media content items, the media application may transmit a scene request, which causes the top scene generator to communicate with the ML model to identify the starting and ending times of the top scenes. Then, the media application may render another user interface (e.g., the item's landing page) that provides additional details about the program and the identification of the top scenes.

[0021] In some examples, the media platform may include a scene ranker that ranks the top scenes identified by the ML model according to one or more user preferences stored in a user account associated with the media application. For example, the scene ranker may rank the order of the top scenes (as they are presented on the user interface) according to the user's preferences. The user preferences may include watch history, user-specified interests, user engagements on the favorability of media content items, and/or other user preferences indicated by the user and/or derived from one or more other ML models inputted with television signals about the user's watch history. In some examples, the top scene generator is stored at the media platform on one or more server computers. In some examples, the top scene generator is stored at the display device. These and other features are further described with reference to the figures.

[0022] FIG. 1 illustrates a system 100 that identifies a scene 121 in a media content item 108 based on a natural language query 168 and/or generates an edited version 126 of a media content item 108 by removing one or more scenes 121 satisfying one or more content restriction categories 172. The system 100 may increase the performance of a display device 152 by reducing latency and/or buffering issues caused by manually navigating through a timeline of a media content item 108 to locate a particular scene 121 of interest to the user. For example, conventionally, a user may move a UI element (e.g., a skip forward button, a skip backward button, a timeline slider) to navigate to different parts of the media content item 108, but navigating back and forward in the item's timeline may cause latency and/or buffering issues because repeatedly moving a current playback point in the timeline may cause the display of content to be delayed due to network latency. However, programmatically navigating to a particular scene 121 using a natural language query 168 may efficiently display and render a scene of interest while avoiding latency and/or buffering issues caused by manually navigating back and/or forth through the item's timeline.

[0023] The system 100 includes a media platform 104 executable by one or more server computers 102 and a media application 156 executable by a display device 152. The media platform 104 may be a server-based television platform. In some examples, the media application 156 is (or is a subcomponent of) an operating system 151 of the display device 152. In some examples, the media application 156 is a native application (e.g., a standalone native application), which is preinstalled on the display device 152 or downloaded to the display device 152 from a digital media store (e.g., play store, application store, etc.). The media application 156 may communicate with the media platform 104 to identify media content 106 that is available for streaming to the display device 152. The media content 106 includes a plurality of media content items 108. In some examples, the media content 106 includes media content items 108 that are stored on the media platform 104 and streamed from the media platform 104 to the media application 156. In some examples, the media content 106 includes media content items 108 that are stored on one or more (other) streaming platforms 128 and streamed from the streaming platforms 128 to their respective streaming applications 154.

[0024] In some examples, the media application 156 is a media aggregator application that determines which providers (e.g., streaming platforms 128, associated streaming applications 154) the user has access rights to, and then identifies media content items 108, across those providers, in the user interface 164 for selection and playback. For example, the media application 156 (e.g., in conjunction with the media platform 104) may aggregate (e.g.,

combine, assemble, collect, etc.) information about media content 106 available for viewing (e.g., streaming) from multiple streaming platforms 128 and present the information in the user interface 164 (e.g., a single, unified user interface) so that a user can identify and/or search media content 106 across different streaming platforms (e.g., without having to search within each streaming application 154). In some examples, the media content 106 is referred to as media content items 108 (e.g., individual programs offered by streaming platforms 128). For example, each media content item 108 may be a program (e.g., a television show, a movie, a live broadcast, etc.) from the media platform 104 or another streaming platform 128. Instead of searching for media content items 108 on a first streaming application and media content items 108 on a second streaming application, the media application 156 may combine the media content items 108 together in one interface (e.g., user interface 164) so that a user can search across multiple streaming platforms 128 at once.

[0025] In some examples, a media content item 108 may correspond to a digital video file, which may be stored on the streaming platforms 128 (including the media platform 104) and/or the display device 152. In some examples, the media platform 104 is also considered a streaming platform 128, which may store and provide digital video files for streaming or downloading. The digital video file may include video and/or audio data that corresponds to a particular media content item 108. In some examples, the media platform 104 is configured to communicate with the streaming platforms 128 to identify which media content 106 is available on the streaming platforms 128 and may update a media provider database 105 to identify the media content items 108 offered by the streaming platforms 128.

[0026] For example, the media platform 104 may communicate, over a network 150, with the streaming platforms 128 to identify which media content 106 is available to be streamed by display devices 152 and update a media provider database 105. The media platform 104 may identify a set or multiple sets of media content items 108 (e.g., across the various streaming platforms 128) as recommendations to a user of the media application 156. In some examples, the media platform 104 may determine whether the user of the media application 156 has rights (e.g., stored as entitlement data 112) to stream media content 106 from one or more of the streaming platforms 128 (e.g., whether the user has subscribed to access media content 106 from the streaming platform(s) 128), and, if so, may include those media content items 108 as candidates in a selection (e.g., ranking) mechanism to potentially be displayed in the user interface 164 of the media application 156.

[0027] The media application 156 includes a user interface 164 that identifies media content items 108 for selection and playback on the display device 152. In response to

selection of a media content item 108, the media application 156 may initiate playback of the media content item 108 on a display 162 of the display device 152. In some examples, in response to selection of the media content item 108, the media platform 104 streams the media content item 108 to the media application 156, which causes the media application 156 to display the media content item 108 on the display 162. In some examples, in response to selection of the media content item 108 from the user interface 164 of the media application 156, the media application 156 causes the content's underlying streaming application 156 to playback the media content item 108.

[0028] In some examples, selection of a media content item 108 from the user interface 164 may cause the media application 156 to launch a streaming application 154 (e.g., using a content deep link) associated with the streaming application 154. In some examples, selection of a media content item 108 from the user interface 164 causes the media application 156 to render another user interface (e.g., item's landing page), and further selection of the media content item 108 from the item's landing page causes the media application 156 to launch the underlying streaming application 154. In some examples, the media content item 108 may be associated with a specific provider in which the media content item 108 is streamed from a streaming platform 128 (e.g., the media platform 104 itself or another streaming platform 128). In some examples, the user can control the playback of the media content item 108 from the corresponding streaming application 154.

[0029] A content deep link, corresponding to a media content item 108, may be an identifier that identifies the location of the media content item 108 in the streaming application 154. The media application 156 may transfer the content deep link to the corresponding streaming application 154. In some examples, the content deep link identifies a specific landing page (e.g., an interface) within the streaming application 154 that corresponds to the media content item 108. In some examples, the content deep link is an operating system intent. In some examples, the content deep link is a uniform resource locator (URL). In some examples, the content deep link includes a URL format.

[0030] Streaming (or playback) of the media content item 108 may refer to the transmission of the contents of a video file (e.g., media assets) from a streaming platform 128 or the media platform 104 to the display device 152 that displays the contents of the video file. In some examples, streaming (or playback) of the media content item 108 may refer to a continuous video stream that is transferred from one place to another place in which a received portion of the video stream is displayed while waiting for other portions of the video stream to be transferred. In some examples, after the media content item 108 is published on

the media platform 104 (e.g., is live), the display device 152 may stream or download the contents of the video file.

[0031] In some examples, the user interface 164 may identify a plurality of media content items 108, which may be selected by the media platform 104 from the media provider database 105 based at least in part on information representing the user's interests and activities (e.g., the user's search queries, search results, previous watch history, purchase history, application usage history, application installation history, user actions on the network-connected display device, physical activities of the user, etc.). In some examples, the media application 156 may be associated with a user account 110, and the user account 110 may store the information representing the user's interests and activities (e.g., user activity information 114), and the media platform 104 may use this information to select and present the media content items 108 in the user interface 164. In some examples, the media content items 108 may be organized as a plurality of clusters based on one or more categories, such as content type (e.g., "Action Movies"), viewing history (e.g., "Because You watched Movie ABC"), release time (e.g., "Trending"), and the like. In some examples, the media content items 108 provided by different streaming platforms 128 (e.g., action movies from two different streaming platforms 128) can be recommended in the same cluster. In some examples, the user interface 164 may include tabbed interfaces, where one of the tabbed interfaces includes personalized media content that is organized as a plurality of clusters based on one or more categories, such as release time (e.g., "This Week," "Next week," "Next Month," etc.), user action and user application interaction, native app usage (e.g., items that are "From App ABC"), etc.

[0032] It is noted that a user of the media application 156 may be provided with controls allowing the user to make an election as to both if and when the system 100 may enable the collection of information representing the user's interests and activities. In addition, certain data may be treated in one or more ways before it is stored or used, so that personally identifiable information is removed. For example, a user's identity may be treated so that no personally identifiable information can be determined for the user, or a user's geographic location may be generalized where location information is obtained (such as to a city, ZIP code, or state level), so that a particular location of a user cannot be determined. Thus, the user of the media application 156 may have control over what information is collected about the user, how that information is used, and what information is provided to the user and/or to the server computer 102.

[0033] The media application 156 displays an interface 166 for receiving a natural language query 168 about one or more scenes 121 in the media content item 108. A scene 121 is a portion of the media content item 108. A scene 121 may be a continuous segment of audiovisual content within a larger video or program. In some examples, the file of the media content item 108 includes metadata that identifies scene change markers (e.g., timestamps), which may be used to segment videos into scenes. In some examples, in response to the media content item 108 being played back, the media application 156 initiates display of the interface 166 on the display 162 of the display device 152. In some examples, the media application 156 provides a control (e.g., a dedicated control, a menu item, etc.), which, when selected, causes the display of the interface 166 on the display 162 of the display device 152. In some examples, the interface 166 includes an input field that enables the user to submit a natural language query 168. In some examples, a user may enter text into the input field to define the natural language query 168. In some examples, the interface 166 may be a user interface configured to receive text via a voice command and may display the text of the voice command in the input field. The natural language query 168 may be text that identifies a specific scene 121 in the media content item 108 (e.g., “rhino scene”). In some examples, the natural language query 168 may be text that identifies a genre such as a fighting scene. However, the natural language query 168 may be any textual information about one or more scenes 121.

[0034] The media application 156 includes a prompt generator 158 configured to generate a prompt 130 to be used as an input to the ML model 120. The prompt 130 may include a content identifier 135 of the media content item 108. The content identifier 135 may uniquely identify the media content item 108 on the media platform 104 and/or the streaming platform 128. In response to submission to the natural language query 168 via the interface 166, the prompt generator 158 may generate a prompt 130 with a request to identify one or more scenes 121 in the media content item 108 using the natural language query 168. In some examples, the prompt 130 includes an instruction to identify at least one of a starting time 134 or an ending time 136 of one or more scenes 121 in the media content item 108 based on the natural language query 168. In some examples, the prompt generator 158 is included on the media platform 104. For example, the media platform 104 receives, over a network 150, the natural language query 168, and the prompt generator 158 generates the prompt 130 for use as an input to the ML model 120.

[0035] The media application 156 may transmit the prompt 130 to a ML model 120. In some examples, the ML model 120 is stored on the server computer(s) 102, which also

executes the media platform 104. In some examples, the ML model 120 is stored on one or more server computers that are different from the server computer(s) 102 that execute the media platform 104. In some examples, the media platform 104 receives the prompt 130 and then transmits the prompt 130 to the ML model 120. In some examples, the ML model 120 is stored (e.g., stored locally) on the display device 152. In some examples, the ML model 120 is included as part of the operating system 151. In response to the prompt 130, the ML model 120 may identify the scene(s) 121 that satisfy the natural language query 168. In some examples, the ML model 120 includes a large language model (LLM) 122. In some examples, the LLM 122 is a conventional large language model (e.g., based on a transformer architecture), adapted to generate text in response to a text prompt provided as input. Such LLMs are trained on a large corpus of publicly available text, e.g., content from public databases and websites.

[0036] The LLM 122 may include any type of pre-trained large language model (LLM) configured to identify scenes 121 in a media content item 108 in response to a prompt 130. The LLM 122 includes weights, where the weights are numerical parameters that the LLM 122 learns during the training process. The weights are used to compute the output (e.g., the prompt response 132) of the LLM 122. In some examples, the LLM 122 includes a pre-trained language model that has been fine-tuned with additional training data (e.g., media content items 108) to identify scenes 121 in different types of media content 106.

[0037] The LLM 122 may receive text input, where the text input includes the prompt 130. The LLM 122 may include a pre-processing engine configured to pre-process the text input. Pre-processing may include converting the text input to individual tokens (e.g., words, phrases, or characters). Pre-processing may include other operations such as removing stop words (e.g., “the”, “and”, “of”) or other terms or syntax that do not impart any meaning to the LLM 122. The LLM 122 includes an embedding engine configured to generate word embeddings from the pre-processed text input. The word embeddings may be vector representations that assist the LLM 122 to capture the semantic meaning of the input tokens and may assist the LLM 122 to better understand the relationships between the input tokens. The LLM 122 includes a neural network(s) configured to receive the word embeddings and generate an output.

[0038] A neural network includes multiple layers of interconnected neurons (e.g., nodes). The neural network may include an input layer, one or more hidden layers, and an output layer. The output may include a sequence of output word probability distributions, where each output distribution represents the probability of the next word in the sequence

given the input sequence so far. In some examples, the output may be represented as a probability distribution over the vocabulary or a subset of the vocabulary. The neural network(s) is configured to receive the word embeddings and generate an output, and, in some examples, the query activity (e.g., previous natural language queries 168 and prompt responses 132). The output may represent a version of the textual response. The output may include a sequence of output word probability distributions, where each output distribution represents the probability of the next word in the sequence given the input sequence so far. In some examples, the output may be represented as a probability distribution over the vocabulary or a subset of the vocabulary. The decoder is configured to receive the output and identify the scenes 121 of a media content item 108. In some examples, the decoder may select the most likely instruction, sampling from a probability distribution, or using other techniques to generate coherent and valid source code. The LLM 122 includes a decoder configured to receive the output and generate a prompt response 132 with the starting time 134 and the ending time 136.

[0039] The ML model 120 generates and transmits a prompt response 132 that includes a starting time 134 and/or an ending time 136 of the scene(s) 121 that satisfy the natural language query 168. In some examples, the prompt response 132 includes information that identifies a particular scene 121 (e.g., “final dogfight scene”). In some examples, the prompt response 132 includes information that identifies a short description of the scene 121. In some examples, the prompt response 132 includes information in JSON notation. The media application 156 includes a playback controller 160 configured to control playback of the media content item 108 based on the prompt response 132. For example, in response to the prompt response 132, the playback controller 160 may programmatically navigate the media content item 108 (or cause a respective streaming application 154 to programmatically navigate the media content item 108) based on the starting time 134 and/or the ending time 136 of the scene(s) 121. In some examples, programmatically navigating includes navigating playback of the media content item 108 to the starting time 134 of the scene 121. In some examples, programmatically navigating includes updating a current playback point to the starting time 134 of the scene 121. In some examples, the playback controller 160 skips to the scene 121 identified in the prompt response 132 in response to a further user instruction. For example, the playback controller 160 may enable the user to search for certain types of scenes (or specific scenes) such as fight scenes, comedy scenes, etc.

[0040] In some examples, programmatically navigating includes, during playback of the media content item 108, skipping the scene 121 based on the starting time 134 and the ending time 136 of the scene 121. For example, the playback controller 160 may implement a fast forward feature that can be used to skip intros, songs, fight scenes, adult scenes, etc. For example, the playback controller 160 may determine whether a current time point in the media content item 108 corresponds to the starting time 134, and, if so, automatically fast forwards to the ending time 136 of the scene 121. In some examples, the prompt response 132 identifies a list of scenes 121 that satisfy the natural language query 168. For example, the scenes 121 may include a scene 121-1, a scene 121-2, and a scene 121-3, each of which satisfies the natural language query 168. The interface 166 may include one or more navigation control elements that enable the user to skip to or go back to the scenes 121 that satisfy the natural language query 168. For example, in response to a navigation control element, the playback controller 160 may navigate to a subsequent scene 121 in the list of scenes 121.

[0041] In some examples, the media platform 104 may programmatically generate edited versions 126 of media content items using the ML model 120. For example, the media platform 104 may include a version generator 124 configured to generate an edited version 126 of a media content item 108. In some examples, the version generator 124 generates an edited version 126 of a media content item 108 when the media content item 108 is published to the media platform 104. In some examples, the version generator 124 generates an edited version 126 of a media content item 108 in response to a natural language query 168 received via the interface 166 (e.g., “create a kid-friendly version of the current program”). In some examples, the version generator 124 generates an edited version 126 of a media content item 108 based on information received via a custom editing interface 170. In some examples, the media platform 104 may generate an edited version 126 of a media content item 108 by communicating with the ML model 120 to identify one or more scenes 121 that include restricted content and editing the one or more scenes 121 to remove the restricted content from playback of the media content item 108.

[0042] The media platform 104 may store user accounts 110, where each user account 110 stores information about a respective user. A user account 110 may store entitlement data 112 and/or user activity information 114. The entitlement data 112 includes information that identifies which providers (e.g., streaming platforms 128, streaming applications 154) that the user account 110 has access rights to view content. In some examples, the access rights are determined based on the user account 110 (e.g., whether the user has subscribed to

one or more streaming applications 154), which streaming applications 154 are installed on the display device 152 and/or if the user has accessed (e.g., logged-into) a user account associated with those streaming applications 154. In response to certain user activity regarding media content items 108, the media platform 104 may update the user activity information 114 with information about the activity such as a content identifier 135, the date/time, and/or the watch duration of the media content item 108, etc.

[0043] In some examples, a user account 110 includes one or more profiles 116. A user account 110 may include multiple profiles such as one or more adult profiles and one or more child profiles. The media platform 104 may define one or more content restriction settings for a user account 110 (e.g., for a particular profile 116). In some examples, a profile 116 may be associated with a content restriction setting 118. In some examples, the content restriction setting 118 is a parental control setting. In some examples, if the content restriction setting 118 is enabled, the media platform 104 may provide media content items 108 that meet a certain rating (e.g., Motion Picture Association of America (MPAA) in the user interface 164, which, in some examples, may include edited versions 126 of media content items 108. For example, if a particular profile 116 has an enabled content restriction setting 118, the media platform 104 may provide the edited version 126 of the media content item 108. In some examples, the non-edited version may be a version that is not eligible to be viewed under the profile 116, but the edited version 126 may be a version that is eligible to be viewed under the profile 116.

[0044] In some examples, the media platform 104 may enable a user to customize an edited version 126 of a media content item 108. For example, the media application 156 may render a custom editing interface 170 with a plurality of content restriction categories 172 with respect to a particular media content item 108, a group of media content items 108, or all the media content items 108 rendered under a particular profile 116. The content restriction categories 172 may be different categories of content that may be deemed unsuitable for viewing such as violence, profanity, sexual content, drug use, and/or scary content. A user may use the custom editing interface 170 to select one or more content restriction categories 172. The media application 156 may generate an editing request 131 to the media platform 104, which causes the media platform 104 to communicate with the ML model 120 to identify and then remove one or more scenes 121 that satisfy the content restriction categories 172 selected by the user.

[0045] The display device 152 includes one or more processors, one or more memory devices, and an operating system 151 configured to execute (or assist with executing) one or

more streaming applications 154. The one or more memory devices may be a non-transitory computer-readable medium storing executable instructions that cause the one or more processors to execute operations discussed herein. The display device 152 may be any type of user device. In some examples, the display device 152 is a television device (e.g., a smart television). In some examples, the display device 152 is a smartphone, a laptop computer, a desktop computer, a gaming console, and/or a wearable device such as a head-mounted display device. In some examples, the display device 152 is an augmented reality (AR) or virtual reality (VR) device. The streaming applications 154 may include a media application 156 configured to communicate, over the network, 150, with a media platform 104 executable by one or more server computers 102. In some examples, the media application 156 is a program that is part of the operating system 151. In some examples, the media application 156 is a separate standalone application that is downloaded and installed on the operating system 151. In some examples, the media application 156 may execute operation(s) discussed with reference to the operating system 151 (and/or vice versa). In some examples, the display device 152 is not a smart television, but is converted to a smart television when connected to a casting device, where the casting device is configured to connect to the network 150 and execute an operating system 151 configured to execute streaming applications 154, including the media application 156.

[0046] In some examples, the operating system 151 is a browser application. A browser application is a web browser configured to access information on the Internet and may launch one or more browser tabs in the context of one or more browser windows. In some examples, the operating system 151 is a Linux-based operating system. In some examples, the operating system 151 is a mobile operating system that is also configured to execute on smaller devices (e.g., smartphones, tablets, wearables, etc.).

[0047] The server computer 102 may be computing devices that take the form of a number of different devices, for example a standard server, a group of such servers, or a rack server system. In some examples, the server computer 102 may be a single system sharing components such as processors and memories. The network 150 may include the Internet and/or other types of data networks, such as a local area network (LAN), a wide area network (WAN), a cellular network, satellite network, or other types of data networks. The network 150 may also include any number of computing devices (e.g., computer, servers, routers, network switches, etc.) that are configured to receive and/or transmit data within network 150. Network 150 may further include any number of hardwired and/or wireless connections.

[0048] The server computer 102 may include one or more processors formed in a substrate, an operating system (not shown) and one or more memory devices. The memory devices may represent any kind of (or multiple kinds of) memory (e.g., RAM, flash, cache, disk, tape, etc.). In some examples (not shown), the memory devices may include external storage, e.g., memory physically remote from but accessible by the server computer 102. The server computer 102 may include one or more modules or engines representing specially programmed software.

[0049] FIG. 2A and 2B illustrate example user interfaces for navigating to a scene using a natural language query 268 submitted via an interface 266 according to an aspect. The example user interfaces of FIGS. 2A and 2B may be examples of user interfaces provided by the system of FIG. 1. As shown in FIG. 2A, a media application may render a user interface 284, which displays playback of a media content item 208. FIG. 2A depicts a scene 221-1 of the media content item 208. The user interface 284 displays an interface 266 for receiving a natural language query 268. In response to the natural language query 268, as shown in FIG. 2B, the media application programmatically navigates to a scene 221-2 that corresponds to the natural language query 268.

[0050] FIG. 3A and 3B illustrate example user interfaces for navigating to a scene using a natural language query 368 submitted via an interface 366 according to an aspect. The example user interfaces of FIGS. 3A and 3B may be examples of user interfaces provided by the system of FIG. 1. As shown in FIG. 3A, a media application may render a user interface 384, which displays playback of a media content item 308. FIG. 3A depicts a scene 321-1 of the media content item 308. The user interface 384 displays an interface 366 for receiving a natural language query 368. In response to the natural language query 368, as shown in FIG. 3B, the media application programmatically navigates to a scene 321-2 that corresponds to the natural language query 368.

[0051] FIGS. 4A to 4C illustrate a streaming system 400 according to another aspect. The streaming system 400 may be an example of the streaming system 100 of FIG. 1 and may include any of the details discussed with reference to those figures. As shown in FIG. 4A, the streaming system 400 includes a media platform 404 and a display device 452. The media platform 404 and the display device 452 may be examples of the media platform 104 and the display device 152 of FIG. 1 and may include any of the details discussed with reference to those figures.

[0052] In some examples, the media platform 404 includes a top scene generator 442 configured to communicate with a ML model 420 to generate top scenes 421 in a media

content item 408 that is available for selection in a user interface 464 of a media application 456. A top scene 421 may be an important, influential, or stand-out portion of a program that is identified by the ML model 420. The ML model 420 may be a pre-trained LLM. The top scene generator 442 configured to generate a prompt 430 for the ML model 420, where the prompt 430 includes an instruction for the ML model 420 to identify top scenes 421 in a particular media content item 408. In some examples, the prompt 430 includes information that identifies a particular media content item 408 (e.g., name of a program) with an instruction to identify the top scenes 421 in the media content item 408. The ML model 420 may identify a starting time 434 and an ending time 436 of each top scene 421 and transmit a prompt response 432 with the starting time 434 and the ending time 436 of each top scene 421. The starting time 434 and the ending time 436 may be timestamps in the program. In some examples, the prompt response 432 also includes a short description of the top scene. The top scene generator 442 may provide the list of top scenes 421 (with their starting time 434 and the ending time 436) to the media application 456, which renders the list of top scenes 421 as selectable items on the user interface 464. As shown in FIG. 4A, the user interface 464 identifies a top scene 421-1, a top scene 421-2, and a top scene 421-3 as the top scenes 421 in the media content item 408. Although three top scenes 421 are identified in FIG. 4A, the number of top scenes 421 may be any number greater than three.

[0053] In some examples, in response to selection of a top scene 421-1 from the user interface 464, the media application 456 causes the media content item 408 to playback from the selected top scene 421-1. In some examples, the media content item 408 continues to play from the end of the top scene 421-1. In some examples, after the selected top scene 421-1 has ended, the media application 456 may re-render the user interface 464 that identifies the top scenes 421. In some examples, after the selected top scene 421-1 has ended, the media application 456 may render an interface that asks the user whether they want to continue with the media content item 408 or return to the top scene selection interface. In some examples, after the selected top scene 421-1 has ended, the media application 456 may cause the playback of the next top scene 421 (e.g., top scene 421-2).

[0054] In some examples, the top scene generator 442 is configured to communicate with the ML model 420 to generate top scenes 421 in real-time. For example, the top scenes 421 in a particular media content item 408 may change over time, and the top scene generator 442 is configured to generate the top scenes 421 in real-time (e.g., when the user requests a media content item's landing page). For example, in response to selection of a media content item 408 among a plurality of media content items (e.g., the selection of a media content item

408 in FIG. 4B), the media application 456 may transmit a scene request 443, which causes the top scene generator 442 to communicate with the ML model 420 to identify the starting time 434 and the ending time 436 of the top scenes 421. Then, the media application 456 may render another user interface (e.g., an interface 464a in FIG. 4C) that provides additional details about the program and the identification of the top scenes 421. In some examples, the interface 464a is a landing page associated with the media content item 408.

[0055] In some examples, the media platform 404 (or the top scene generator 442) may include a scene ranker 448 that ranks the top scenes 421 identified by the ML model 420 according to one or more user preferences 444 stored in a user account 410 associated with the media application 456. For example, the scene ranker 448 may rank the order of the top scenes 421 (as they are presented on the user interface 464) according to the user preferences 444. The user preferences 444 may include watch history, user-specified interests, user engagements on the favorability of media content items, and/or other user preferences indicated by the user and/or derived from one or more other ML models inputted with television signals about the user's watch history. In some examples, the top scene generator 442 is stored at the media platform 404 on one or more server computers. In some examples, the top scene generator 442 is stored at the display device 452.

[0056] FIG. 5 is a flowchart 500 depicting example operations of a system that identifies a scene in a media content item based on a natural language query and/or generates an edited version of a media content item by removing one or more scenes satisfying one or more content restriction categories 172. The flowchart 500 may depict operations of a computer-implemented method. The flowchart 500 may depict operations of a non-transitory computer-readable medium having executable instructions that when executed by one or more processors cause the one or more processors to execute the operations of the flowchart 500. Although the flowchart 500 is explained with respect to the system 100 of FIG. 1, the flowchart 500 may be applicable to any of the implementations discussed herein, including the streaming system 400 of FIG. 4. Although the flowchart 500 of FIG. 5 illustrates the operations in sequential order, it will be appreciated that this is merely an example, and that additional or alternative operations may be included. Further, operations of FIG. 5 and related operations may be executed in a different order than that shown, or in a parallel or overlapping fashion.

[0057] Operation 502 includes initiating display of a media content item on a display device. Operation 504 includes receiving, via an interface of a media application, a natural language query about at least one scene in the media content item. Operation 506 includes, in

response to the natural language query, generating, by the media application, a prompt including a request to identify the at least one scene using the natural language query. Operation 508 includes receiving, from a machine-learning (ML) model, a prompt response that identifies at least one of a starting time or an ending time of the at least one scene. Operation 510 includes programmatically navigating, by the media application, the media content item based on at least one of the starting time or the ending time of the at least one scene.

[0058] Clause 1. A method comprising: initiating display of a media content item on a display device; receiving, via an interface of a media application, a natural language query about at least one scene in the media content item; in response to the natural language query, generating, by the media application, a prompt including a request to identify the at least one scene using the natural language query; receiving, from a machine-learning (ML) model, a prompt response that identifies at least one of a starting time or an ending time of the at least one scene; and programmatically navigating, by the media application, the media content item based on at least one of the starting time or the ending time of the at least one scene.

[0059] Clause 2. The method of clause 1, wherein the programmatically navigating includes: navigating playback of the media content item to the starting time of the at least one scene.

[0060] Clause 3. The method of clause 1 or 2, wherein the programmatically navigating includes: during playback of the media content item, skipping the at least one scene based on the starting time and the ending time of the at least one scene.

[0061] Clause 4. The method of any one of clauses 1 to 3, wherein the interface includes a navigation control, the prompt response identifying a list of scenes that satisfy the natural language query, the method further comprising: in response to selection of the navigation control, navigating to a subsequent scene in the list of scenes.

[0062] Clause 5. The method of any one of clauses 1 to 4, further comprising: generating an edited version of the media content item, including: communicating with the ML model to identify one or more scenes that include restricted content; and editing the one or more scenes to remove the restricted content from playback of the media content item.

[0063] Clause 6. The method of clause 5, further comprising: determining that a content restriction setting is enabled with respect to a user profile of the media application; and in response to the content restriction setting being enabled, providing the edited version of the media content item for selection under the user profile.

[0064] Clause 7. The method of any one of clauses 1 to 6, further comprising: rendering, by the media application, a custom editing interface with a plurality of content restriction categories; receiving a selection to a content restriction category of the plurality of content restriction categories; and generating an editing request to the ML model, the editing request instructing the ML model to remove one or more scenes that satisfy the content restriction category.

[0065] Clause 8. A computer program product storing executable instructions that when executed by one or more processors cause the one or more processors to execute any one of clauses 1 to 7.

[0066] Clause 9. An apparatus comprising: at least one processor; and a non-transitory computer-readable medium storing executable instructions that cause the at least one processor to execute operations, the operations comprising: initiating display of a media content item on a display device; receiving, via an interface of a media application, a natural language query about at least one scene in the media content item; in response to the natural language query, generating, by the media application, a prompt including a request to identify the at least one scene using the natural language query; receiving, from a machine-learning (ML) model, a prompt response that identifies at least one of a starting time or an ending time of the at least one scene; and programmatically navigating, by the media application, the media content item based on at least one of the starting time or the ending time of the at least one scene.

[0067] Clause 10. The apparatus of clause 9, wherein the programmatically navigating includes: navigating playback of the media content item to the starting time of the at least one scene.

[0068] Clause 11. The apparatus of clause 9 or 10, wherein the programmatically navigating includes: during playback of the media content item, skipping the at least one scene based on the starting time and the ending time of the at least one scene.

[0069] Clause 12. The apparatus of any one of clauses 9 to 11, wherein the interface includes a navigation control, the prompt response identifying a list of scenes that satisfy the natural language query, the operations further comprising: in response to selection of the navigation control, navigating to a subsequent scene in the list of scenes.

[0070] Clause 13. The apparatus of any one of clauses 9 to 12, further comprising: generating an edited version of the media content item, including: communicating with the ML model to identify one or more scenes that include restricted content; and editing the one or more scenes to remove the restricted content from playback of the media content item.

[0071] Clause 14. The apparatus of clause 13, further comprising: determining that a content restriction setting is enabled with respect to a user profile of the media application; and in response to the content restriction setting being enabled, providing the edited version of the media content item for selection under the user profile.

[0072] Clause 15. The apparatus of any one of clauses 9 to 15, wherein the operations further comprise: rendering, by the media application, a custom editing interface with a plurality of content restriction categories; receiving a selection to a content restriction category of the plurality of content restriction categories; and generating an editing request to the ML model, the editing request instructing the ML model to remove one or more scenes that satisfy the content restriction category.

[0073] Clause 16. A method comprising: generating, by a top scene generator, a prompt including a request to identify a plurality of top scenes in a media content item available for selection from a media platform; receiving, from a machine-learning (ML) model, a prompt response that identifies a starting time and an ending time of each of the plurality of top scenes; and initiating display of a list of the plurality of top scenes for selection on a user interface of a display device; and in response to selection of a top scene from list, initiating playback of the media content item from the starting time.

[0074] Clause 17. The method of clause 16, further comprising: initiating display of a plurality of media content items, including the media content item; in response to selection of the media content item from the user interface, generating a scene request; and in response to the scene request, transmitting the prompt to the ML model.

[0075] Various implementations of the systems and techniques described here can be realized in digital electronic circuitry, integrated circuitry, specially designed ASICs (application specific integrated circuits), computer hardware, firmware, software, and/or combinations thereof. These various implementations can include implementation in one or more computer programs that are executable and/or interpretable on a programmable system including at least one programmable processor, which may be special or general purpose, coupled to receive data and instructions from, and to transmit data and instructions to, a storage system, at least one input device, and at least one output device.

[0076] These computer programs (also known as programs, software, software applications or code) include machine instructions for a programmable processor and can be implemented in a high-level procedural and/or object-oriented programming language, and/or in assembly/machine language. As used herein, the terms “machine-readable medium” “computer-readable medium” refers to any computer program product, apparatus and/or

device (e.g., magnetic discs, optical disks, memory, Programmable Logic Devices (PLDs)) used to provide machine instructions and/or data to a programmable processor, including a machine-readable medium that receives machine instructions as a machine-readable signal. The term “machine-readable signal” refers to any signal used to provide machine instructions and/or data to a programmable processor.

[0077] To provide for interaction with a user, the systems and techniques described here can be implemented on a computer having a display device (e.g., a CRT (cathode ray tube) or LCD (liquid crystal display) monitor) for displaying information to the user and a keyboard and a pointing device (e.g., a mouse or a trackball) by which the user can provide input to the computer. Other kinds of devices can be used to provide for interaction with a user as well; for example, feedback provided to the user can be any form of sensory feedback (e.g., visual feedback, auditory feedback, or tactile feedback); and input from the user can be received in any form, including acoustic, speech, or tactile input.

[0078] The systems and techniques described here can be implemented in a computing system that includes a back end component (e.g., as a data server), or that includes a middleware component (e.g., an application server), or that includes a front end component (e.g., a client computer having a graphical user interface or a Web browser through which a user can interact with an implementation of the systems and techniques described here), or any combination of such back end, middleware, or front end components. The components of the system can be interconnected by any form or medium of digital data communication (e.g., a communication network). Examples of communication networks include a local area network (“LAN”), a wide area network (“WAN”), and the Internet.

[0079] The computing system can include clients and servers. A client and server are remote from each other and typically interact through a communication network. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship with each other.

[0080] In this specification and the appended claims, the singular forms “a,” “an” and “the” do not exclude the plural reference unless the context clearly dictates otherwise. Further, conjunctions such as “and,” “or,” and “and/or” are inclusive unless the context clearly dictates otherwise. For example, “A and/or B” includes A alone, B alone, and A with B. Further, connecting lines or connectors shown in the various figures presented are intended to represent example functional relationships and/or physical or logical couplings between the various elements. Many alternative or additional functional relationships, physical connections or logical connections may be present in a practical device. Moreover,

no item or component is essential to the practice of the implementations disclosed herein unless the element is specifically described as “essential” or “critical”.

[0081] Terms such as, but not limited to, approximately, substantially, generally, etc. are used herein to indicate that a precise value or range thereof is not required and need not be specified. As used herein, the terms discussed above will have ready and instant meaning to one of ordinary skill in the art.

[0082] Moreover, use of terms such as up, down, top, bottom, side, end, front, back, etc. herein are used with reference to a currently considered or illustrated orientation. If they are considered with respect to another orientation, it should be understood that such terms must be correspondingly modified.

[0083] Although certain example methods, apparatuses and articles of manufacture have been described herein, the scope of coverage of this patent is not limited thereto. It is to be understood that terminology employed herein is for the purpose of describing particular aspects and is not intended to be limiting. On the contrary, this patent covers all methods, apparatus and articles of manufacture fairly falling within the scope of the claims of this patent.

WHAT IS CLAIMED IS:

1. A method comprising:
 - initiating display of a media content item on a display device;
 - receiving, via an interface of a media application, a natural language query about at least one scene in the media content item;
 - in response to the natural language query, generating, by the media application, a prompt including a request to identify the at least one scene using the natural language query;
 - receiving, from a machine-learning (ML) model, a prompt response that identifies at least one of a starting time or an ending time of the at least one scene; and
 - programmatically navigating, by the media application, the media content item based on at least one of the starting time or the ending time of the at least one scene.
2. The method of claim 1, wherein the programmatically navigating includes:
 - navigating playback of the media content item to the starting time of the at least one scene.
3. The method of claim 1 or 2, wherein the programmatically navigating includes:
 - during playback of the media content item, skipping the at least one scene based on the starting time and the ending time of the at least one scene.
4. The method of any one of claims 1 to 3, wherein the interface includes a navigation control, the prompt response identifying a list of scenes that satisfy the natural language query, the method further comprising:
 - in response to selection of the navigation control, navigating to a subsequent scene in the list of scenes.
5. The method of any one of claims 1 to 4, further comprising:
 - generating an edited version of the media content item, including:
 - communicating with the ML model to identify one or more scenes that include restricted content; and
 - editing the one or more scenes to remove the restricted content from playback of the media content item.

6. The method of claim 5, further comprising:
 - determining that a content restriction setting is enabled with respect to a user profile of the media application; and
 - in response to the content restriction setting being enabled, providing the edited version of the media content item for selection under the user profile.
7. The method of any one of claims 1 to 6, further comprising:
 - rendering, by the media application, a custom editing interface with a plurality of content restriction categories;
 - receiving a selection to a content restriction category of the plurality of content restriction categories; and
 - generating an editing request to the ML model, the editing request instructing the ML model to remove one or more scenes that satisfy the content restriction category.
8. A computer program product storing executable instructions that when executed by one or more processors cause the one or more processors to execute any one of claims 1 to 7.
9. An apparatus comprising:
 - at least one processor; and
 - a non-transitory computer-readable medium storing executable instructions that cause the at least one processor to execute operations, the operations comprising:
 - initiating display of a media content item on a display device;
 - receiving, via an interface of a media application, a natural language query about at least one scene in the media content item;
 - in response to the natural language query, generating, by the media application, a prompt including a request to identify the at least one scene using the natural language query;
 - receiving, from a machine-learning (ML) model, a prompt response that identifies at least one of a starting time or an ending time of the at least one scene; and
 - programmatically navigating, by the media application, the media content item based on at least one of the starting time or the ending time of the at least one scene.

10. The apparatus of claim 9, wherein the programmatically navigating includes:
navigating playback of the media content item to the starting time of the at least one scene.
11. The apparatus of claim 9 or 10, wherein the programmatically navigating includes:
during playback of the media content item, skipping the at least one scene based on the starting time and the ending time of the at least one scene.
12. The apparatus of any one of claims 9 to 11, wherein the interface includes a navigation control, the prompt response identifying a list of scenes that satisfy the natural language query, the operations further comprising:
in response to selection of the navigation control, navigating to a subsequent scene in the list of scenes.
13. The apparatus of any one of claims 9 to 12, further comprising:
generating an edited version of the media content item, including:
communicating with the ML model to identify one or more scenes that include restricted content; and
editing the one or more scenes to remove the restricted content from playback of the media content item.
14. The apparatus of claim 13, further comprising:
determining that a content restriction setting is enabled with respect to a user profile of the media application; and
in response to the content restriction setting being enabled, providing the edited version of the media content item for selection under the user profile.
15. The apparatus of any one of claims 9 to 15, wherein the operations further comprise:
rendering, by the media application, a custom editing interface with a plurality of content restriction categories;
receiving a selection to a content restriction category of the plurality of content restriction categories; and
generating an editing request to the ML model, the editing request instructing the ML model to remove one or more scenes that satisfy the content restriction category.

16. A method comprising:
- generating, by a top scene generator, a prompt including a request to identify a plurality of top scenes in a media content item available for selection from a media platform;
 - receiving, from a machine-learning (ML) model, a prompt response that identifies a starting time and an ending time of each of the plurality of top scenes; and
 - initiating display of a list of the plurality of top scenes for selection on a user interface of a display device; and
 - in response to selection of a top scene from list, initiating playback of the media content item from the starting time.
17. The method of claim 16, further comprising:
- initiating display of a plurality of media content items, including the media content item;
 - in response to selection of the media content item from the user interface, generating a scene request; and
 - in response to the scene request, transmitting the prompt to the ML model.

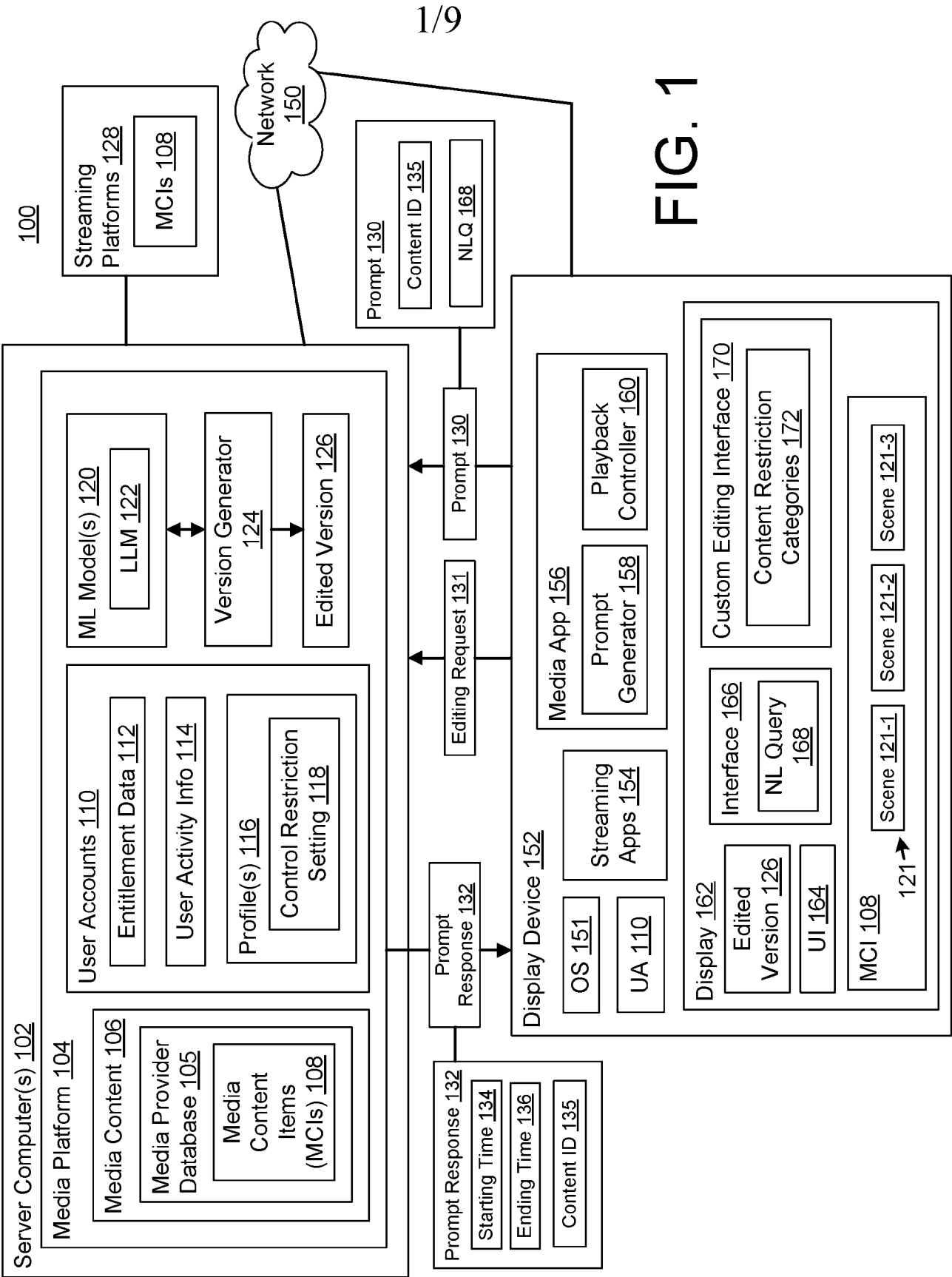


FIG. 1

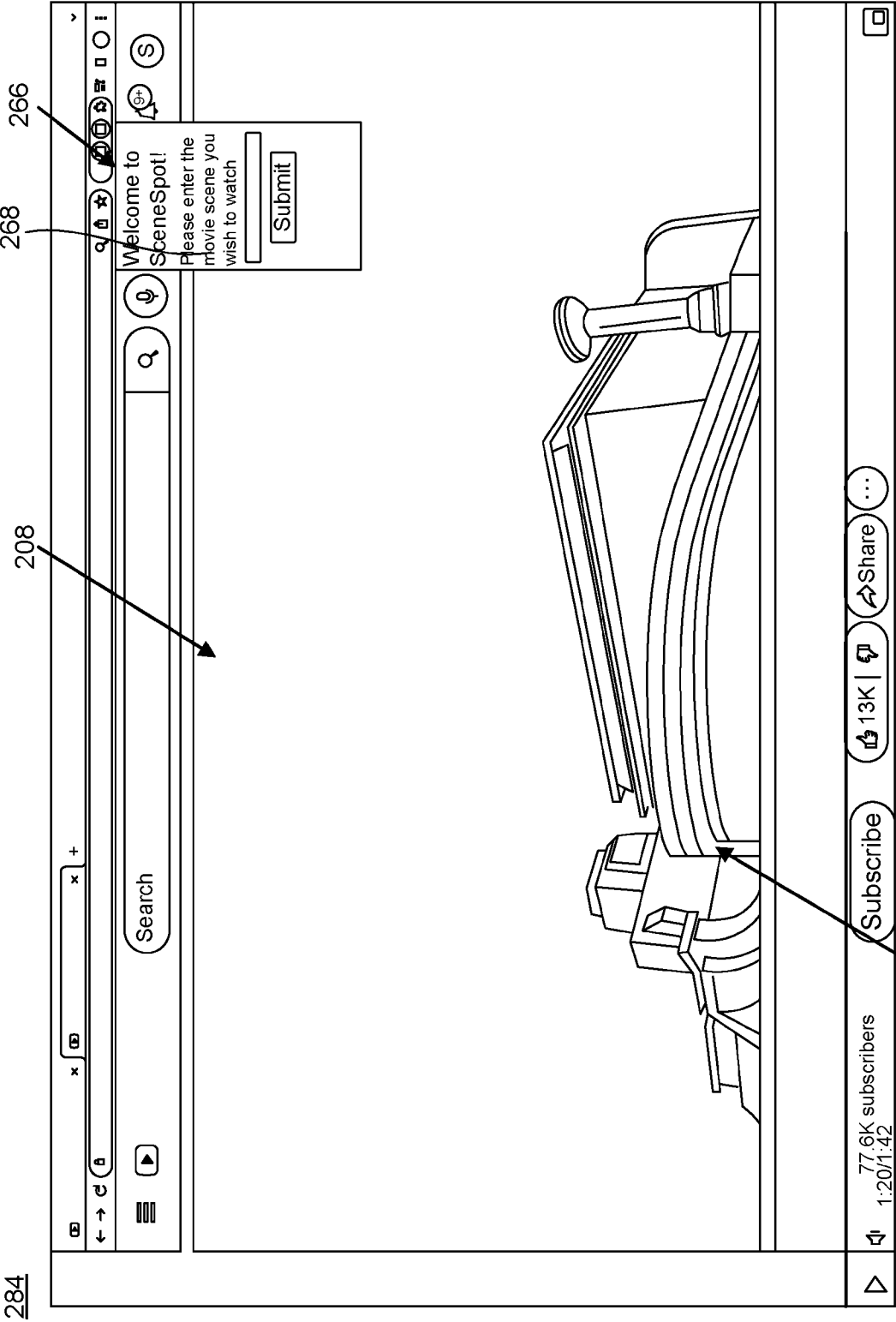
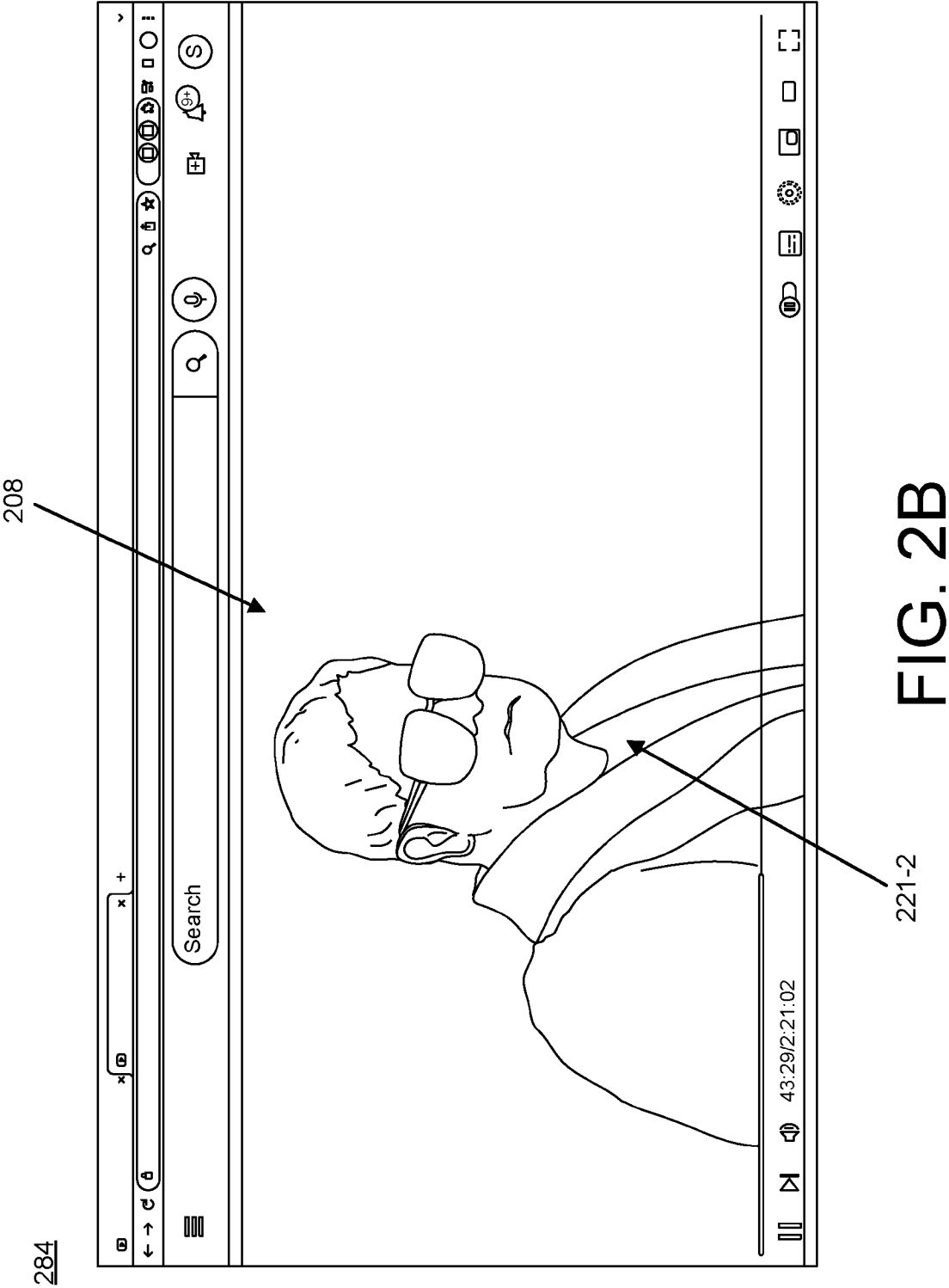


FIG. 2A



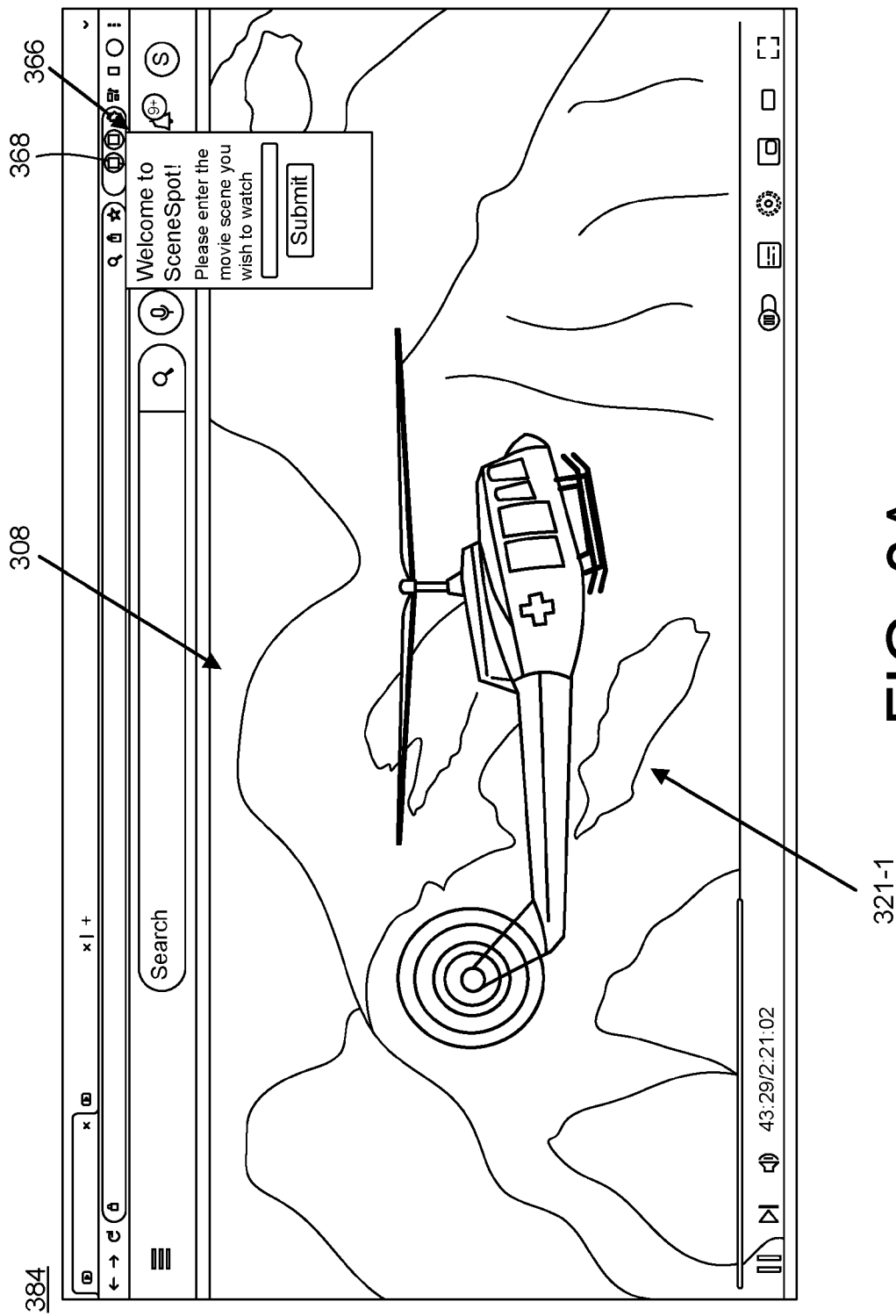


FIG. 3A

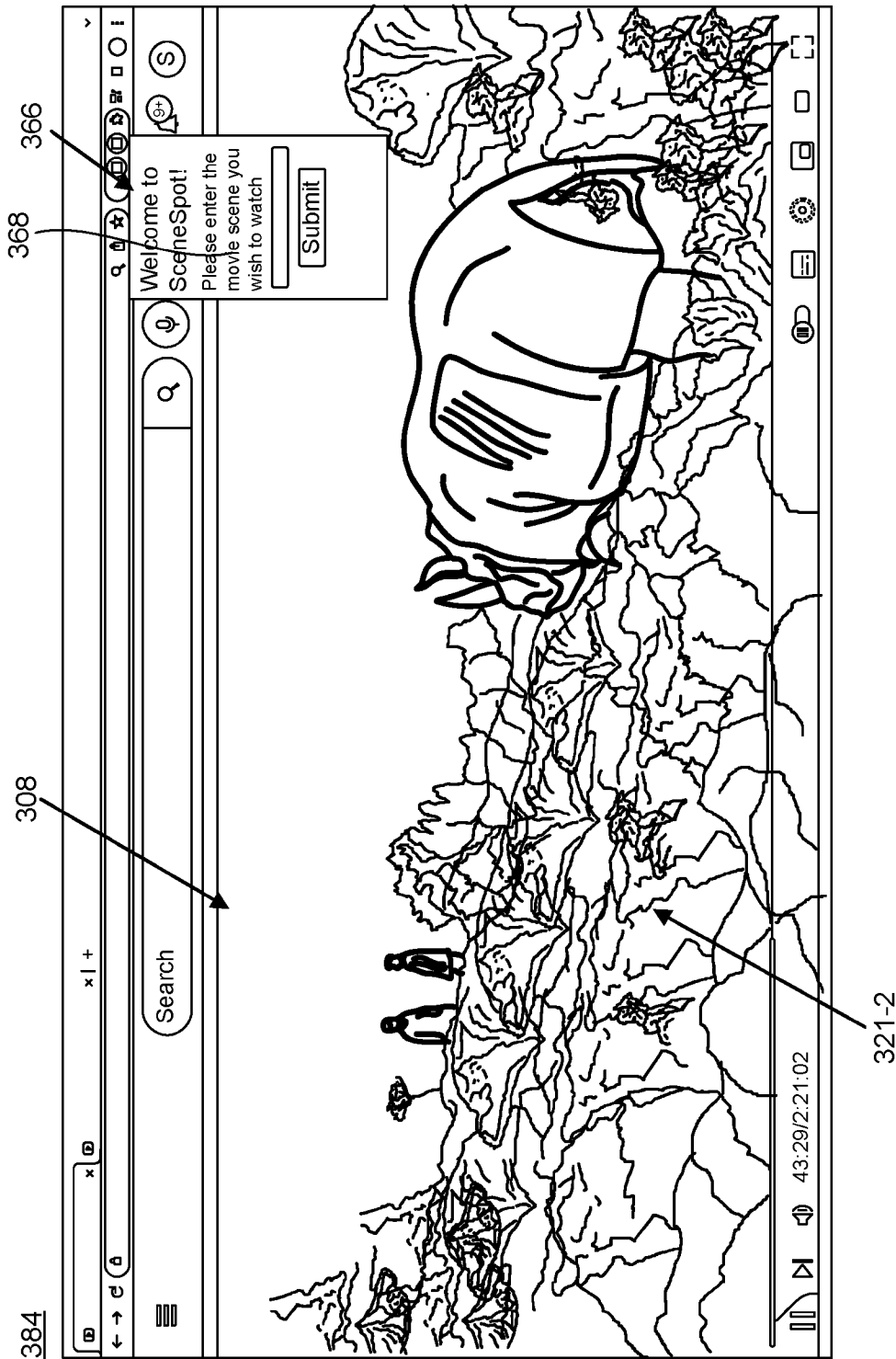
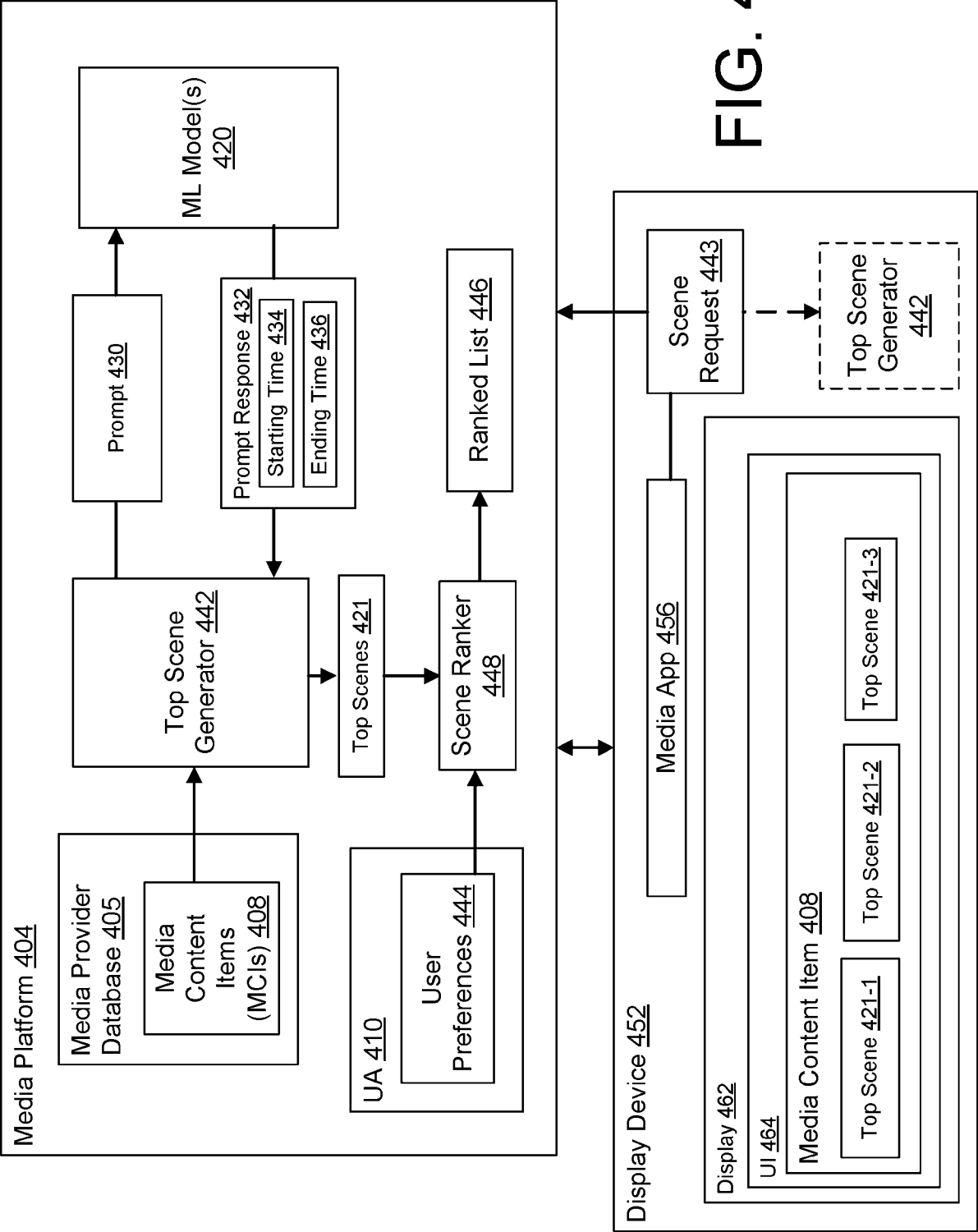


FIG. 3B

400



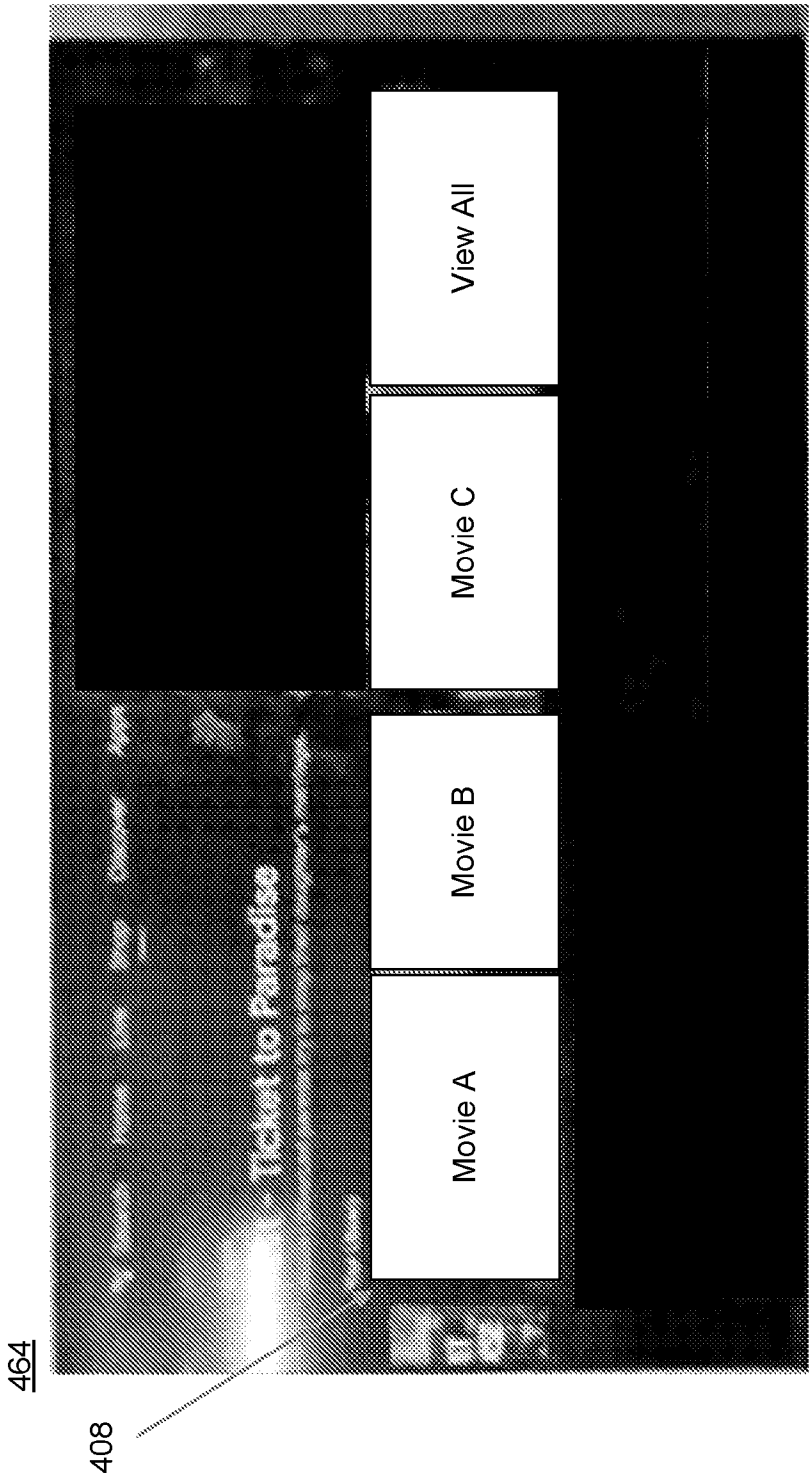


FIG. 4B

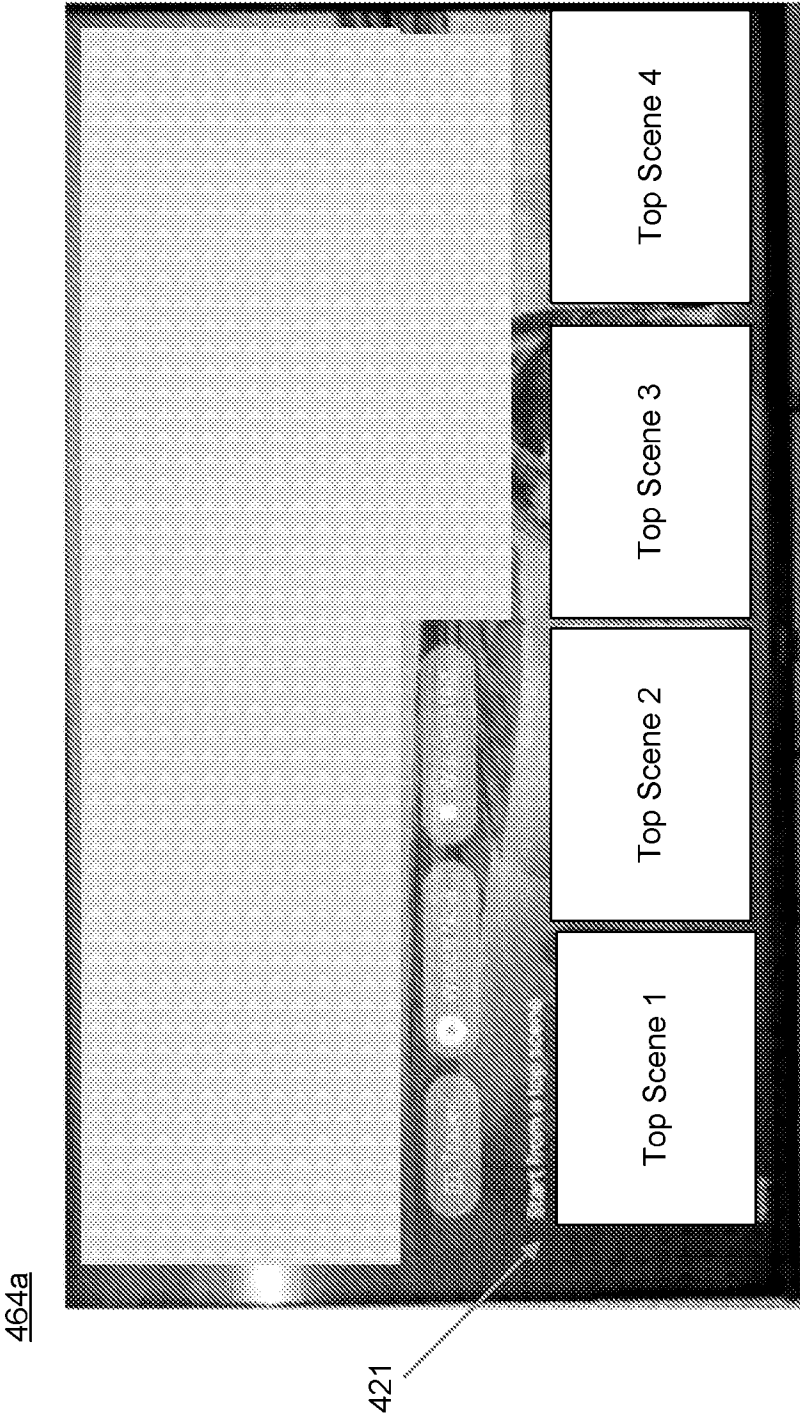


FIG. 4C

9/9

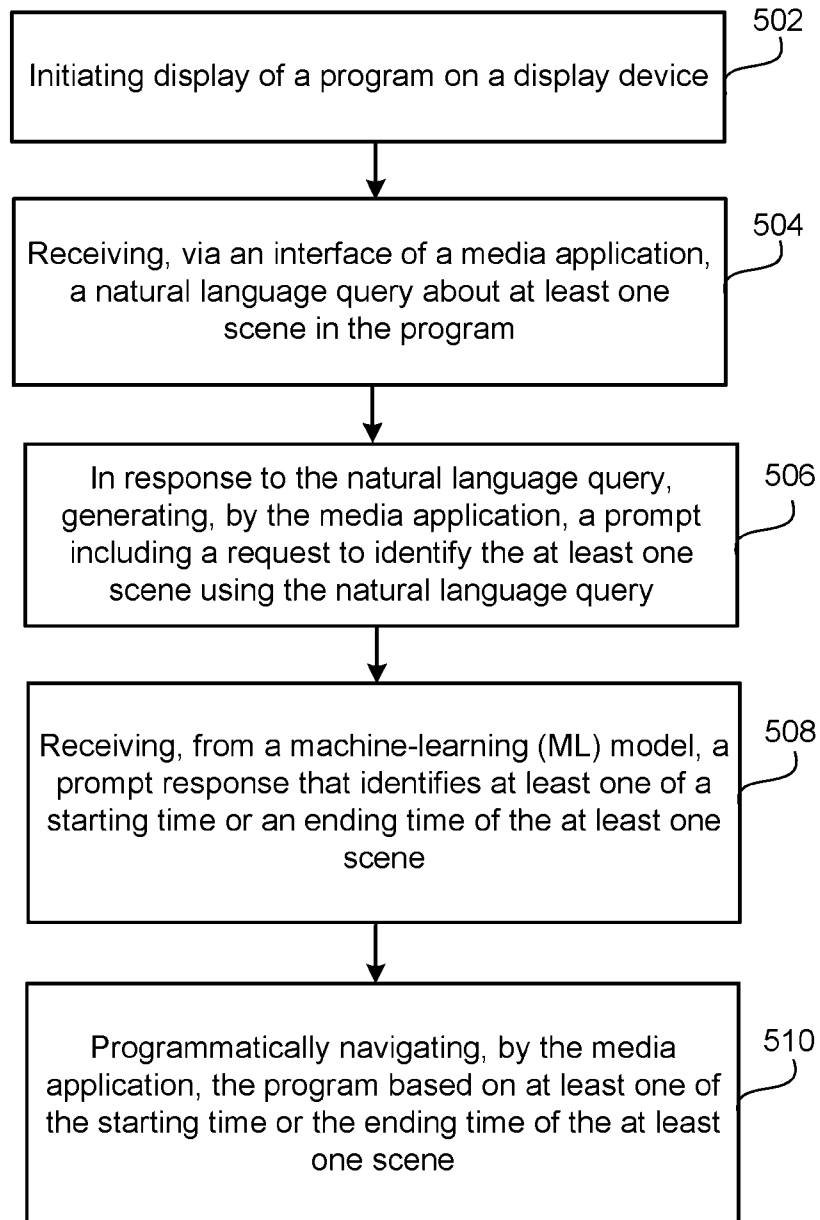
500

FIG. 5

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2023/084864

A. CLASSIFICATION OF SUBJECT MATTER

INV. G06F16/78

ADD.

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI Data**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2019/354608 A1 (DEPAZ ALEXANDER [CA] ET AL) 21 November 2019 (2019-11-21) abstract paragraphs [0004] - [0006], [0039] - [0041]; figures 2,3 -----	1 - 17
X	US 11 789 992 B2 (TIVO CORP [US]) 17 October 2023 (2023-10-17) abstract column 2, line 46 - column 3, line 50 column 5, line 66 - column 6, line 13 -----	1 - 17
X	US 2011/047163 A1 (CHECHIK GAL [US] ET AL) 24 February 2011 (2011-02-24) abstract paragraphs [0007] - [0009], [0028], [0062]; figure 10 ----- - / - -	1 - 17



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance;; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance;; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

23 May 2024

Date of mailing of the international search report

17/06/2024

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2

NL - 2280 HV Rijswijk

Tel. (+31-70) 340-2040,

Fax: (+31-70) 340-3016

Authorized officer

Siódsmok, Wojciech

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2023/084864

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2018/025078 A1 (QUENNESSON KEVIN [US]) 25 January 2018 (2018-01-25) abstract paragraphs [0003], [0008], [0041], [0042], [0068] - [0071], [0080], [0085]; figure 6A -----	1 - 17

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2023/084864

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2019354608 A1	21-11-2019	NONE	
US 11789992 B2	17-10-2023	US 2013290846 A1	31-10-2013
		US 2017371871 A1	28-12-2017
		US 2020233892 A1	23-07-2020
		US 2023418860 A1	28-12-2023
US 2011047163 A1	24-02-2011	AU 2010286797 A1	15-03-2012
		AU 2016202074 A1	28-04-2016
		AU 2018201624 A1	29-03-2018
		CA 2771593 A1	03-03-2011
		CN 102549603 A	04-07-2012
		EP 2471026 A1	04-07-2012
		EP 3352104 A1	25-07-2018
		US 2011047163 A1	24-02-2011
		US 2015220543 A1	06-08-2015
		US 2018349391 A1	06-12-2018
		US 2021349944 A1	11-11-2021
		US 2023306057 A1	28-09-2023
		WO 2011025701 A1	03-03-2011
US 2018025078 A1	25-01-2018	EP 3488618 A1	29-05-2019
		US 2018025078 A1	25-01-2018
		WO 2018017836 A1	25-01-2018