Technical Disclosure Commons

Defensive Publications Series

June 2022

ENABLING TOUCH CAPABILITIES ON AUTO DASHBOARD SCREENS WITHOUT TOUCHSCREENS

D. Shin

Follow this and additional works at: https://www.tdcommons.org/dpubs_series

Recommended Citation
Shin, D., "ENABLING TOUCH CAPABILITIES ON AUTO DASHBOARD SCREENS WITHOUT TOUCHSCREENS", Technical Disclosure Commons, (June 27, 2022)
https://www.tdcommons.org/dpubs_series/5223

This work is licensed under a Creative Commons Attribution 4.0 License.
This Article is brought to you for free and open access by Technical Disclosure Commons. It has been accepted for inclusion in Defensive Publications Series by an authorized administrator of Technical Disclosure Commons.

ENABLING TOUCH CAPABILITIES ON AUTO DASHBOARD SCREENS WITHOUT TOUCHSCREENS

ABSTRACT

A vehicle (e.g., automobile, motorcycle, a bus, a recreational vehicle (RV), a semi-trailer truck, a tractor or other type of farm equipment, a train, a plane, a helicopter, etc.) may have displays to make visual presentations of useful information such as maps, personalized routes, media, etc. Often a separate computing device (e.g., a smartphone, a laptop computer, a tablet computer, smartglasses, a portable gaming system, a laptop computer, a smartwatch, etc.) may provide the visual presentation data for the vehicle. Newer model vehicles typically have touchscreen displays, but some vehicles may have displays without an associated touchscreen. In those cases, the user may be required to modify the content on the dashboard through the computing device, which may increase the interaction time and may be cumbersome.

In this disclosure, the computing device may use its camera to determine user interactions with the non-touchscreen display of the vehicle. The camera of the computing device may be pointed toward the non-touchscreen display, and the computing device may apply a perspective correction algorithm with initial user calibration input of dashboard corner zones. A hand tracking model at the computing device may then determine the position of a user's hand with respect to the non-touchscreen display of the vehicle. A determination of hand contact with the touchscreen may be done using a machine learning model, such as a machine learning calibrated embedding model. The computing device may then compute dashboard screen coordinates for hand contact at the non-touch screen device and update the visual presentation at the non-touchscreen display appropriately.

DESCRIPTION

FIG. 1 below is a conceptual diagram of a system 10 including a head unit 100 of a vehicle (e.g., an automobile, a motorcycle, a bus, a recreational vehicle (RV), a semi-trailer truck, a tractor or other type of farm equipment, a train, a plane, a helicopter, etc.) and a computing device 120 (e.g., a smartphone, a tablet computer, smartglasses, a smartwatch, a portable gaming system, a laptop computer, etc.). As shown in FIG. 1, head unit 100 includes one or more processors 102, a display 104, one or more communication components 106 (“COMM components 106”), and one or more storage devices 112. As further shown in FIG. 2, computing device 120 includes one or more processors 122, a display 124, one or more communication components 126 (“COMM components 126”), and one or more storage devices 132.

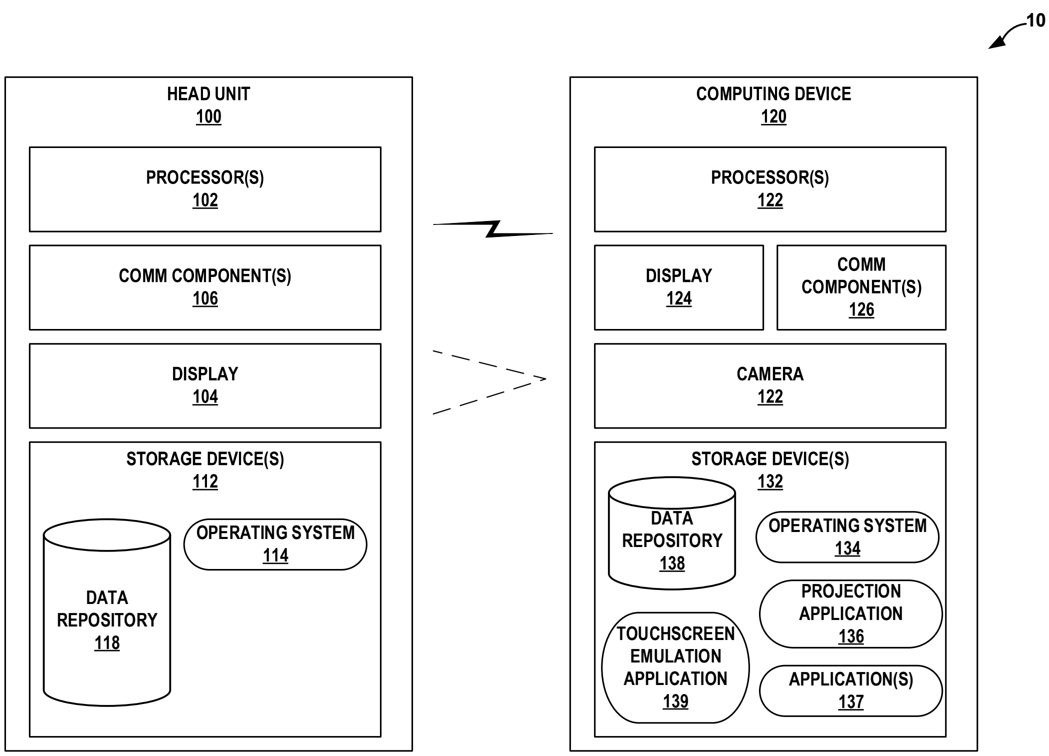

FIG. 1

One or more components (e.g., processors 102, display 104, COMM components 106, storage devices 112, etc.) of head unit 100 may be substantially similar to one or more components (e.g., processors 122, display 124, COMM components 126, storage devices 132, etc.) of second computing device 120. As such, the description of one may apply equally to the other except for any differences described herein.

Head unit 100 of system 10 may operate to assist, inform, entertain, or otherwise provide for interactions with one or more occupants of a vehicle. Head unit 100 may represent an integrated head unit that provides a user interface (UI), such as a voice user interface (VUI), a graphical user interface (GUI), etc. In general, head unit 100 may control one or more vehicle systems, such as a heating, ventilation, and air conditioning (HVAC) system, a lighting system (for controlling interior and/or exterior lights), an infotainment system, a seating system (for controlling a position of a driver and/or passenger seat), etc. Head unit 100 may be configured to establish a session with a computing device to permit data exchange. In some examples, an occupant of the vehicle in which vehicle head unit 100 is located may connect computing device 120 to head unit 100 to project (or otherwise, cast or stream) a UI to head unit 100. For instance, a UI model in head unit 100 may be a thin client that supports projection of a GUI from computing device 120.

Processors 102 may implement functionality and/or execute instructions associated with head unit 100. Examples of processors 102 may include one or more of an application specific integrated circuit (ASIC), a field programmable gate array (FPGA), an application processor, a display controller, an auxiliary processor, a central processing unit (CPU), a graphics processing unit (GPU), one or more sensor hubs, and any other hardware configure to function as a processor, a processing unit, or a processing device. Processors 102 may retrieve and execute instructions stored by storage devices 112 that cause processors 102 to perform the operations described in this disclosure.

Display104 may function as an output (e.g., display) device using any of one or more display components, such as a liquid crystal display (LCD), dot matrix display, light emitting diode (LED) display, active-matrix organic light-emitting diode (AMOLED) display, etc. As discussed below, display 104 of head unit 100 may be a non-presence-sensitive display that has limited or no functionality as an input device.

COMM components 106 of head unit 100 may include wireless communication devices capable of transmitting and/or receiving communication signals, such as a cellular radio, a 3G radio, a 4G radio, a 5G radio, a Bluetooth® radio (or any other PAN radio), an NFC radio, or a Wi-Fi™ radio (or any other wireless local area network (WLAN) radio). COMM components 106 may be configured to send and receive information via a network (e.g., a local area network (LAN), wide area network (WAN), a global network, such as the Internet, etc.).

Storage devices 112 of head unit 100 may include one or more computer-readable storage media. For example, storage devices 112 may be configured for long-term, as well as short-term storage of information, such as instructions, data, or other information used by head unit 100. In some examples, storage devices 112 may include non-volatile storage elements. Examples of such non-volatile storage elements include magnetic hard discs, optical discs, solid state discs, etc. Examples of volatile memory devices include random-access memories (RAM), dynamic random-access memories (DRAM), static random-access memories (SRAM), etc.

As shown in FIG. 1, storage devices 112 may include an operating system 114 (“OS 114”) that provides an execution environment for one or more applications, such as applications

137. OS 114 may represent a multi-threaded operating system or a single-threaded operating system with which a projection application 136 may interface to access hardware of head unit 100. OS 114 may include a kernel that facilitates access to the underlying hardware of head unit 100, where the kernel may present a number of different interfaces (e.g., application programmer interfaces – APIs) that projection application 136 may invoke to access the underlying hardware of head unit 100.

Projection application 136 may represent an application that provides a bridge between applications 137 and facilitates projection of UIs, such as GUIs, VUIs, etc., for applications 137 to head unit 100. In some examples, projection application 136 may provide, manage, update, and/or control the UIs. For instance, projection application 136 may provide via display 104 a GUI that, for example, includes information to the user in the form of text, images, etc. Projection application 136 may generate graphical elements to, for example, satisfy driver distraction standards and accommodate a variety of car screen factors and input modalities. Actuation of graphical elements may invoke corresponding functions of applications 137.

Computing device 120 may use touchscreen emulation application 139 to convert conventional, no-touch-input dashboard screens into an interactive touch screen using camera 122. For example, as shown in FIG. 2 below, computing device 120 may be docked near the vehicle's center armrest with camera 122 having display 104 in its field-of-view.

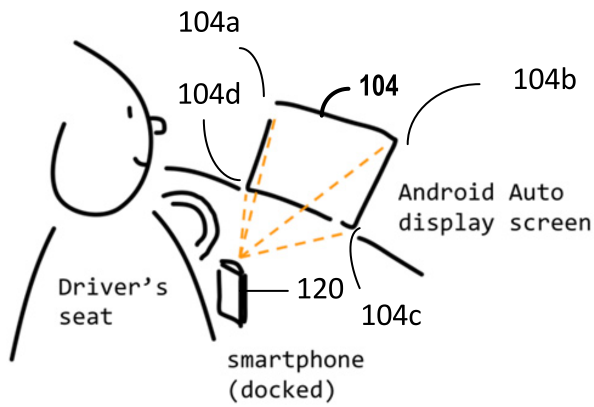

FIG. 2

As shown in FIG. 3 below, to account for non-ideal setups, touchscreen emulation application 139 may use a smart perspective correction algorithm with initial user calibration input indicating the corners of display 104.

Touchscreen emulation application 139 may use a sparse hand tracker model to track the user's hand and determine a hand-to-screen correspondence. During feature setup, touchscreen emulation application 139 may instruct the user to pinpoint the four corners 104a, 104b,104c and 104d (as shown in figure FIG. 22) of the display 104 in a viewfinder shown in the display 124 of computing device 120.

Touchscreen emulation application 139 may determine the dashboard screen coordinates for the touch and send the dashboard screen coordinates to the display 104 using projection application 136 to trigger an appropriate animation. Thus, from the user's perspective, the display 104, which may provide limited or no touch-sensitive capability (in terms of hardware and/or software), may be turned into a proxy touch screen, enabling direct and faster user interactions. Further, by enabling such proxy touch screen capabilities, the operator of the vehicle may more safely interact with display 104 of head unit 100 rather than attempt to interact with display 124 of computing device 120 (which may even be in violation of various laws and regulation in certain jurisdictions) to perform operations (such as updating navigation, changing music, controlling the heating, venting and air conditioning system, etc.).

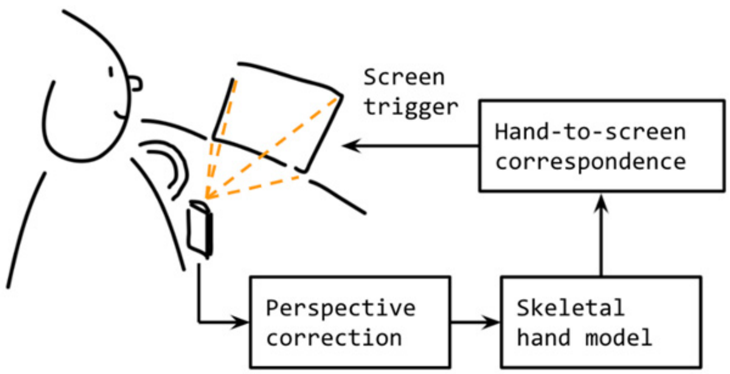

FIG. 3

Figure 4 below shows four corner calibration to perform perspective correction. The perspective corrected image will give a grid view of the target display and effectively remove image variations coming from phone-camera-to display positional misalignment.

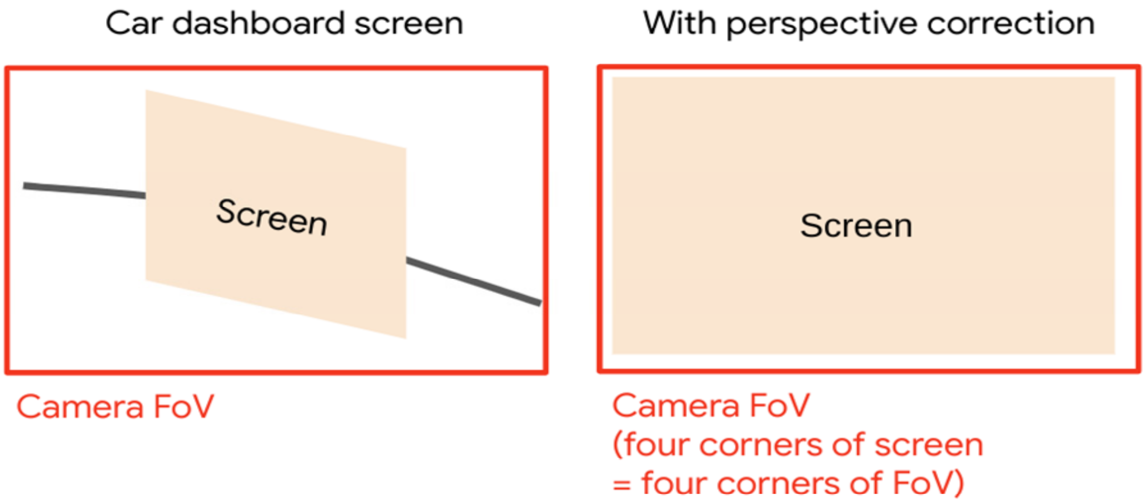

FIG. 4

Figure 5 below shows a representation of a sparse hand tracker model to track the user's hand.

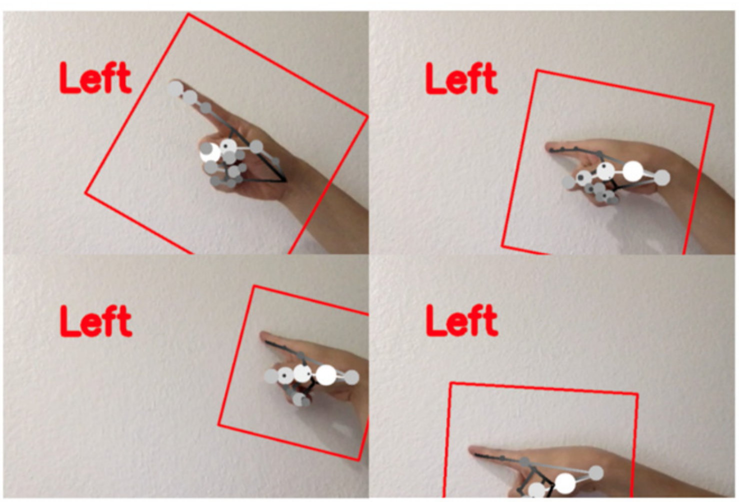

FIG. 5

Using the sparse hand tracker model, touchscreen emulation application 139 may differentiate between a hovering finger, and the finger touching the target display and only register a valid touch event when the user's finger touches the display.

Touchscreen emulation application 139 may determine, again according to the sparse hand tracker model, whether a display touch has occurred by comparing the embedding of the reconstructed hand with a few touch calibration results collected during initial use.

As shown in figure 6 below, touchscreen emulation application 139 may use machine learning calibrated embedding models 602, 604, and 606 to compare current images (or sets of images) from camera 122 initial calibration image(s). The embedding model may produce embeddings which are dense numerical representations of real-world objects and relationships expressed as a vector. Embedding vectors that are close to each other are considered similar.

Touchscreen emulation application 139 may use an error function, such as the L2 loss function, between hand embeddings from each incoming frame with the hand embeddings of stored calibration embedding results. Touchscreen emulation application 139 may determine a valid touch event if the error function between the embedding vector for the image from camera and the embedding vector for a calibrated touch is lower than the error function between the embedding vector for the image from the camera and the embedding vector for a calibrated hovering finger image.

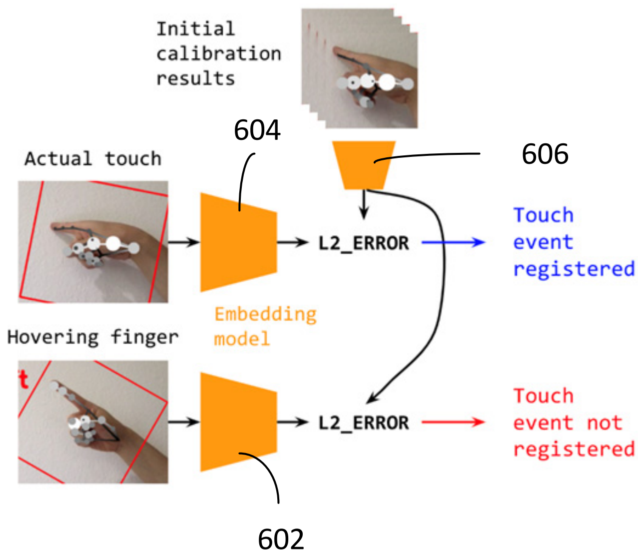

FIG. 6

As shown in FIG. 7 below, touchscreen emulation application 139 may localize fingertip keypoint coordinates in (x, y) plane space once a touch is detected.

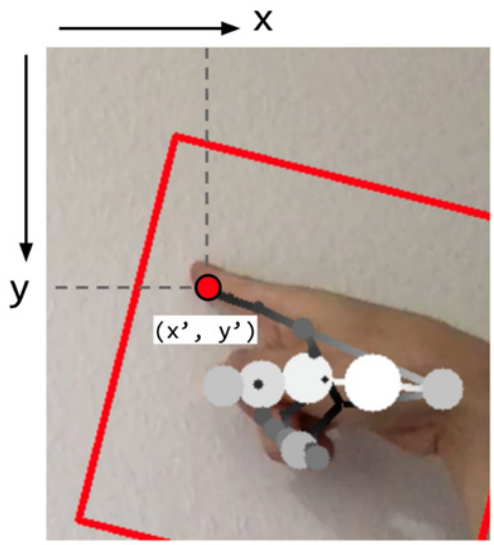

FIG. 7

The touchscreen emulation application 139 may overlay the keypoint coordinates in (x,y) plane space on the perspective corrected car display image to determine the selected user interface element. The touchscreen emulation application 139 may determine any gestures of the hand that emulate touchscreen routines. (e.g., swipe, pinch to zoom, etc.) The touchscreen emulation application 139 may infer touchscreen routines by tracking the finger coordinates over time and building a trajectory dataset. Similar to single finger contact operation discussed above, the touchscreen emulation application 139 may also detect multiple finger contacts. Thus, the touchscreen emulation application 139 may emulate the entire input repertoire defined by a conventional touchscreen without a touchscreen embedded in display 114.

It is noted that the techniques of this disclosure may be combined with any other suitable technique or combination of techniques. As one example, the techniques of this disclosure may be combined with the techniques described in U.S. Patent Application Publication No. 2020/0256692 A1. In another example, the techniques of this disclosure may be combined with the techniques described in U.S. Patent Application Publication No. 2017/0313248 A1. In yet another example, the techniques of this disclosure may be combined with the techniques described in French patent document FR 3106913 A1.