Method and System for Remote Control Using Augmented Analogue-Based Visual Image Through 2-Way Interactive Communication

Field of invention

This invention relates to a method and a system of communication through video conferencing applications. More particularly, this invention relates to a method and a system of communication by incorporating commands onto video images and sending the video images through video conferencing applications. Still more particularly, this invention relates to a method and a system of controlling a device from a remote location through video conferencing applications by superimposing patterns of images onto video images sent to control devices at a remote end.

Background

With the increased usage of messaging over the Internet, there have been prior efforts in leveraging on communication tools (for example, Instant Messaging, MSN, etc.) for controlling of devices remotely via the use of the communication tools.

One such possible effort in leveraging on communication tool is to extract text from an Instant Messaging Chat in MSN. However, this is greatly dependent on the availability of the Application Programming Interface (API) provided by MSN. Thus, if the communication tool does not allow a third party to extract text digitally from a text message received via the communication tool, a command cannot be sent via text message to remotely control devices.

Thus, those skilled in the art are constantly striving to provide an improved method and system for controlling a device from a remote location.

Summary of Invention

The above and other problems are solved and an advance in the art is provided by a method and a system of communication through video conferencing applications in accordance with this invention. A first advantage of a system and a

method in accordance with embodiments of this invention is that as the communication is performed by leveraging on video conferencing applications. Hence, it can be easily implemented using currently available systems. A second advantage in accordance with embodiments of this invention is that security of the message being conveyed over a network is increases because it is more difficult for video images to be compromised when conveyed over the network. A third advantage in accordance with embodiments of this invention is that it provides real time feedback on the status of devices being controlled remotely.

In accordance with embodiments of this invention, a system and a method for communication using a video conferencing application is provided in the following manner. A command is receive from an input from a user. An encoded command is generated for the received command. The encoded command is then superimposed onto a specific portion of an image received from an image capturing device to form a video message. The video message is subsequently transmitted to a selected virtual driver. In turn, the virtual driver transmits the video message to the video conferencing application.

In accordance with some embodiments of this invention, at a receiving end, a video message is received. The system then monitors for a change in the specific portion in the video message. A command is deciphered from the video message in response to a detected change in the specific portion of the video message. A command message is generated in response to deciphering the command. The command message is transmitted to a controller for controlling a device.

In accordance with some embodiments of this invention, the encoded command is generated in the following manner. The command received is read from the memory. The command is encoded to 8-bit binary code to form a message string. The message string is then stored in the memory as encoded command. A Start and an End character are inserted into the message string. Preferably, the Start character is inserted to a beginning of the message string and the End character is

inserted to an end of the message string. Preferably, the Start character is ~ and the End character is ^.

In accordance with some embodiments of this invention, the encoded command is superimposed onto a specific portion of an image received from an image capturing device in the following manner. A first 8 bits in the message string is read. The first 8 bits is inserted into an overlay. 7 random bits are added to the overlay. A toggle bit is added to the overlay. The overlay is then stored in the memory as an overlay string. In accordance with some of these embodiments, a next 8 bits in the message string is read. The next 8 bits is inserted into a next overlay. 7 random bits are added into the next overlay. A toggle bit is added into the next overlay. The next overlay is then appended to end of the overlay string in the memory.

In accordance with some embodiments of this invention, the overlay is created in the following manner. A box is created for each bit in the 8 bits, 7 random bits and toggle bit in the overlay string. Each of the bits that are a "0" bit in the overlay string is determined. The box of each of the bit determined to be "0" bit is filled with a first predetermined colour. Each of the bits that are a "1" bit in the overlay string is determined. The box of each of the bit determined to be "1" bit is filled with a second predetermined colour. A background in a third predetermined colour is generated for the overlay. The boxes of the 8 bits, 7 random bits, and toggle bit are superimposed onto the background to create the overlay. Preferably, the first predetermined colour is green, the second predetermined colour is red, and the background is black in colour.

In accordance with some embodiments of this invention, the command is superimposed onto the specific portion of the image overlay in the following manner. A first overlay is read from the overlay string in the memory. An image captured by an image capturing device is retrieved. The first overlay is then superimposed onto the specific portion of the image captured by the image capturing device to form the

video message. In accordance with some of these embodiments, a next overlay is read from the overlay string in the memory. A next image captured by said image capturing device is retrieved. The next overlay is then superimposed onto the specific portion of the next image captured by the image capturing device to form the video message. The system repeats reading the next overlay from the overlay string, retrieving the next image captured by the image capturing device and superimposing the next overlay onto the next image for the rest of overlays in the overlay string in the memory.

In accordance with some embodiments of this invention, the change in the specific portion in the video message is monitored in the following manner. An image of the video content received from the video conferencing application is captured. The specific portion from the captured image is extracted. A change in a toggle bit in the specific portion of the captured image is determined. In accordance with some of these embodiments, the system repeats capturing an image of video content, extracting the specific portion and determining a change in the toggle bit in response to a subsequent change in the toggle bit.

In accordance with some embodiments of this invention, the determining of the change in a toggle bit in the specific portion of the captured image is performed in the following manner. A number of image bits in the specific portion is determined. The number of image bits is compared with a predetermined number. The image bits are stored in an image bit string in the memory in response to the number of image bits being identical to the predetermined number. In accordance with some of these embodiments, the specific portion is discarded from the memory in response to the number of image bits is not identical to the predetermined number. Preferably, the predetermined number is 16. In accordance with some of these embodiments, the image bit string is read and a binary bits string is generated based on the image bit string, wherein the image bits in a first predetermined colour are assigned as bit 0 and image bits in a second predetermined colour are assigned as bit 1. Preferably, the first predetermined colour is green and the second predetermined colour is red.

In accordance with some embodiments of this invention, the system and method of receiving said video message in response to change in said specific portion in said video image is performed in the following manner. The binary bits string in the memory is read. A first 16 bits from the binary bits string is extracted and a character based on a first 8 bits of the first 16 bits is determined. The character is then stored to a character string in the memory. In accordance with some of these embodiments, a next 16 bits from the binary bits string is extracted. A character based on a first 8 bits of the next 16 bits is determined. The character is appended the character string in the memory. In accordance with some of these embodiments, the system and method repeat extracting a next 16 bits, determining a character based on a first 8 bits of the next 16 bits, and appending the character to the character string in the memory for the rest of the binary bits string. In accordance with some of these embodiments, a first character in the character string in the memory is read. A determination is performed on whether the first character is a predetermined Start character. If the first character is the predetermined Start character, the first character is removed from the character string. A last character in the character string in the memory is read. A determination is performed on whether the last character is a predetermined End character. If the last character is the predetermined End character, the last character is removed from the character string. Preferably, the Start character is ~ and the End character is ^.

In accordance with some embodiments of this invention, the system and method of deciphering the command from the video message is performed in the following manner. The character string is read from the memory. A command in association to the character string is determined. A command message in association to the determined command is generated.

Brief description of drawings

The above and other features and advantages of a method and a system in accordance with this invention are described in the following detailed description and are shown in the following drawings:

Figure 1 illustrating devices and systems communicatively connected to a network to provide or perform a system and/or method in accordance with this invention;

Figure 2a illustrating an image being captured by an image capturing device in accordance with this invention;

Figure 2b illustrating an image of the screen of a tablet in accordance with an embodiment of this invention;

Figure 2c illustrating a video message being created by the system in accordance with an embodiment of this invention;

Figure 3 illustrating an example of an overlay in accordance with an embodiment of this invention;

Figure 4 illustrating a processing system of a device performing processes to provide a system in accordance with an embodiment of this invention;

Figure 5 illustrating a flow diagram for transmitting a command in accordance with this invention;

Figure 6 illustrating a flow diagram for receiving a command in accordance with this invention;

Figure 7 illustrating a flow diagram for encoding the command to form a message string in accordance with this invention;

Figure 8 illustrating a flow diagram for creating overlays in accordance with this invention;

Figure 9 illustrating a flow diagram for superimposing the overlays onto an image captured by an image capturing device in accordance with this invention;

Figure 10 illustrating a flow diagram for extracting information from a video message in accordance with this invention;

Figure 11 illustrating a flow diagram for determining the information in the video message in accordance with this invention;

Figure 12 illustrating a flow diagram for determining a command based on the determined information in accordance with this invention.

Detailed description

This invention relates to a method and a system of communication through video conferencing applications. More particularly, this invention relates to a method and a system of communication by incorporating commands onto video images and sending the video images through video conferencing applications. Still more particularly, this invention relates to a method and a system of controlling a device from a remote location through video conferencing applications by superimposing patterns of images onto video images to control devices at a remote end.

It is envisioned that a system and/or method in accordance with embodiments of this invention may be used to provide remote control and real-time sensors status feedback via video conferencing applications using patterns of images that are correlated with a predefined binary sequence superimposed on images over video content. The superimposed image can be retrieved on the receiving end with unique image recognition algorithm to provide the relevant command and sensors status information for control devices.

Figure 1 illustrates system 100 including a computer system 110, an image capturing device 120 and a controller 130. Consumer devices such as lighting 141, air conditioner 142, stove 143, television 144, etc., are connected to controller 130. Computer system 110 is a typical processing system such as a desktop computer, laptop computer, or other computer terminal that is connected to a network 190 via either a wired or wireless connection to communicate with another computer system 150, tablet 160 or mobile device 170. Computer systems 110 and 150, tablet 160 and mobile device 170 execute applications that perform the required processes in accordance with this invention. Computer system 150, tablet 160 and mobile device 170 each has a image capturing device 151, 161 and 171 respectively in order to perform video conferencing with computer system 110. Briefly, computer system 110

is in communication with computer system 150, tablet 160 or mobile device 170 via video conferencing applications such as those offered by Skype, Google, MSN, etc. Image capturing device 120 captures the images of consumer devices and transmits to computer system 110. The images captured by image capturing device 120 are then streamed to computer system 150, tablet 160 or mobile device 170. In video conferencing applications, image capturing devices 120, 151, 161 and 171 are typically selected as the camera driver to receive video content and subsequently stream the video content to the other party through the video conferencing applications. However, a virtual camera driver is selected to receive and stream video content to the video conferencing applications. The virtual driver is selected to receive video content from image capturing devices 120, 151, 161 and 171 and overlays created. An overlay is superimposed on an image selected from the video content. The superimposed image is then inserted back to the video content to form a video message.

A user, at a location remote to computer system 110, is able to determine the status of the consumer devices based on the video images received by computer system 150, tablet 160 or mobile device 170. Assuming the air conditioner 142 is in operation and user wishes to switch off air conditioner 142, the user may send a command by incorporating the command in an image of a video content and transmitting the video content having the command over the network 190 to computer system 110. In turn, computer system 110 may transmit the command to controller 130 to switch off air conditioner 142 or cut off the power supply of air conditioner 142.

One skilled in the art will recognize that computer system 150 may also be connected to another controller and video capturing device for monitoring or controlling consumer devices at a location remote to computer system 110 without departing from this invention.

Figures 2a, 2b and 2c illustrate an example of remotely controlling consumer devices performed by system 100. Figure 2a shows an image 210 captured by

image capturing device 120. In figure 2a, the television 144 is switched on. The image 210 is transmitted over network 190. Figure 2b shows the image 210 being received by tablet 160. Figure 2b further shows image 230 which is an image captured by image capturing device 161 being displayed on the screen of tablet 160. Assuming the user wishes to switch off the television 144, he may select icon 220. One skilled in the art will recognize that other methods of selection of command may be used without departing from this invention. Upon the selection of icon 220, an application in the tablet 160 will be executed to send a command to computer system 110. The application being executed will be described further with reference to figures 5 and 7-9 below.

Figure 2c shows a video message 250 being created by tablet 160 to be transmitted to computer system 110. The video message shown in figure 2c is streamed to computer system 110 through a virtual camera driver selected in tablet 160. Video message 250 includes image 230 and overlay 240. Overlay 240 contains the selected command and is superimposed on a specific portion of image 230. In figure 2c, overlay 240 is superimposed on the top right corner of image 230. One skilled in the art will recognize that overlay 240 may also be superimposed over other areas in image 230 as long as the computer system 110 is being programmed to know where to extract overlay 240, and the exact area in image 230 being allocated for overlay 240 to be superimposed on is left as a design choice for those skilled in the art.

Overlay 240 provides a specific command to be sent to computer system 110. The video message 250 is being transmitted over network 190 and received by computer system 110. An application in the computer system 110 is then executed to determine the command and transmit the appropriate command to controller 130. Controller 130 then switches off television 144. The exemplary embodiment of an application being executed is described further with reference to figures 6 and 10-12 below.

Figure 3 shows overlay 240. Overlay 240 includes a frame 244, image bits 241, 242 and 243. In this example, 16 image bits are used. The first 8 image bits 241 contain the alphabet character, as the encoding scheme used is American Standard Code for Information Interchange (ASCII). The last image bit 243 acts as a toggle bit to indicate a next character being received. In the described embodiment, the reserve image bits 242 are random bits that may be used for indicating the time of the character being sent from computer system 150, tablet 160 or mobile device 170. Image bits 241, 242 and 243 are represented in green as a "0" bit and in red as a "1" bit. Further, image bits 241, 242 and 243 are superimposed on a background that is a specific colour. The background colour chosen should be easily contrasted with the colours for the bits. In the described embodiments, the background is black in colour. One skilled in the art will recognize that "0" bit colour, "1" bit colour, and background colour should be different from each other. . Hence, other colour configurations may be used without departing from the invention.

One skilled in the art will recognize that the number of bits in image bits 241 is dependent on the encoding scheme selected. Hence, one skilled in the art will recognize that the number of bits in image bits 241 selected is left as a design choice for those skilled in the art.

Figure 4 illustrates an exemplary processing system 400 that represents the processing systems in computer systems 110 and 150, tablet 150 and mobile device 170 that execute instructions to perform the processes described below in accordance with this invention. One skilled in the art will recognize that the instructions may be stored and/or performed as hardware, firmware, or software without departing from this invention. One skilled in the art will recognize that the exact configuration of each processing system may be different and the exact configuration executing processes in accordance with this invention may vary and processing system 400 shown in Figure 4 is provided by way of example only.

Processing system 400 includes Central Processing Unit (CPU) 405. CPU 405 is a processor, microprocessor, or any combination of processors and microprocessors that execute instructions to perform the processes in accordance with the present invention. CPU 405 connects to memory bus 410 and Input/Output (I/O) bus 415. Memory bus 410 connects CPU 405 to memories 420 and 425 to transmit data and instructions between the memories and CPU 405. I/O bus 415 connects CPU 405 to peripheral devices to transmit and receive data between CPU 405 and the peripheral devices. One skilled in the art will recognize that I/O bus 415 and memory bus 410 may be combined into one bus or subdivided into many other busses and the exact configuration is left to those skilled in the art.

A non-volatile memory 420, such as a Read Only Memory (ROM), is connected to memory bus 410. Non-volatile memory 420 stores instructions and data needed to operate various sub-systems of processing system 400 and to boot the system at start-up. One skilled in the art will recognize that any number of types of memory may be used to perform this function.

A volatile memory 425, such as Random Access Memory (RAM), is also connected to memory bus 410. Volatile memory 425 stores the instructions and data needed by CPU 405 to perform software instructions for processes such as the processes for providing a system in accordance with this invention. One skilled in the art will recognize that any number of types of memory may be used to provide volatile memory and the exact type used is left as a design choice to those skilled in the art.

I/O device 430, keyboard 435, display 440, memory 445, network device 450 and any number of other peripheral devices connect to I/O bus 415 to exchange data with CPU 405 for use in applications being executed by CPU 405. I/O device 430 is any device such as controller 130 and image capturing device 120 that transmits and/or receives data from CPU 405. Keyboard 435 is a specific type of I/O that receives user input and transmits the input to CPU 405. Display 440 receives display data from CPU 405 and display images on a screen for a user to see. Memory 445 is

a device that transmits and receives data to and from CPU 405 for storing data to a media. Network device 450 connects CPU 405 to a network for transmission of data to and from other processing systems.

Figure 5 illustrates a flow diagram of process 500 performed by a system in computer system 150, tablet 160 and mobile device 170 in accordance with an embodiment of this invention. The system is provided by processes stored in software, firmware, or hardware. Process 500 begins in step 505 by monitoring for an input from user. In step 510, process 500 receives a command in response to an input from user. Further steps involved for step 510 will be described further with reference to figures 7 and 8 below. After receiving the command, the command will be translated into image bits over an overlay and superimposed onto an image captured by image capturing devices 151, 161 or 171 to form a video message in step 515. Further steps involved in step 515 will be described further with reference to figure 9 below. The video message will then be transmitted to a virtual camera driver in step 520. The virtual camera driver is a driver selected by user in computer device 150, tablet 160 or mobile device 170 to be used for receiving and transmitting video content over network 190. Video contents are: 1) images captured by image capturing devices 151, 161 or 171; or 2) video message 250 generated by computer system 150, tablet 160 or mobile device 170.

Figure 6 illustrates a flow diagram of process 600 performed by a system in computer system 110 in accordance with an embodiment of this invention. The system is provided by processes stored as software, firmware, or hardware. Process 600 begins in step 605 by monitoring for a change in image being received. In particular, process 600 monitors for a change in the last image bit 243 in overlay 240 which indicates an incoming character being received. In step 610, process 600 receives a video message in response to a change in last image bit 243. Further steps involved for step 610 is described further with reference to figures 10 and 11 below. After receiving the video message, the command in the video message is deciphered in step 615. Further steps involved for step 615 will be described further

with reference to figure 12 below. A command message is then generated and transmitted from computer system 110 to controller 130. Controller 130 then performs the command. Examples of the commands include, but are not limited to, deactivating or activating the relevant consumer devices.

Figure 7 illustrates a flow diagram of process 700 performed by the system in computer system 150, tablet 160 and mobile device 170 in accordance with embodiments of this invention. Process 700 is an embodiment of a process for performing step 510 in process 500. Process 700 begins in step 705 by reading the command received. Commands are predetermined codes for controlling consumer devices and may be in a form of a single character or a string of characters. For example, the command to activate television 144 may be "tv-1" while the command to deactivate television 144 may be "tv-0". A table for storing a translation of the commands may be stored in the memory of computer systems 110 and 150, tablet 160 and mobile device 170.

In step 710, the command is encoded to 8-bit binary codes to form a message string. One such encoding scheme is American Standard Code for Information Interchange (ASCII). One skilled in the art will recognize that other encoding schemes may be used without departing from the invention. Further, the number of binary bits used is dependent on the encoding scheme selected. Hence, one skilled in the art will also recognize that the number of binary bits selected is left as a design choice for those skilled in the art.

In step 715, process 700 inserts a Start character in 8-bit binary codes at the beginning of a message string. In step 720, process 700 inserts an End character in 8-bit binary codes at the end of message string. The Start and End characters are preferably characters that are not commonly used. For example, the Start character may be "~" while the End character may be "^". The use of Start and End characters are to indicate the start and end of message string. One skilled in the art will

recognize that other characters may be used for Start and End characters without departing from this invention.

Message string is then stored in memory in step 725. Process 700 ends after step 725.

Figure 8 illustrates a flow diagram of process 800 performed by the system in computer system 150, tablet 160 and mobile device 170 in accordance with embodiments of this invention. Process 800 is an embodiment of a process performing step 515 in process 500. Process 800 begins in step 805 by reading a first character of the message string. As the first character is the Start character (i.e. "~"), the first character in the message string would be "01111110". In step 810, a toggle bit is added. An overlay is then created for the first character in step 815. An example of an overlay is shown in figure 3 where the first 8 image bits 241 represents the first character, "01111110" and the last image bit 243 represents the toggle bit. An additional 7 bits are optionally added between first character and toggle bit, which are represented as reserve image bits 242 in figure 3. The image bits are represented as boxes as shown in figure 3. All "0" bits are represented as green boxes while all "1" bits are represented as red boxes. Further, the boxes are superimposed on a background. In this embodiment, the background colour is black in colour. One skilled in the art will recognize that "0" bits, "1" bits, and background are different colours from each other. Hence, other colour configurations may be used without departing from the invention.

The overlay is then stored as overlay string in memory in step 820. Process 800 determines if there are any more characters in character string in step 825. Thus, process 800 repeats the steps of creating an overlay for each of the character in message string. If a next character is not present, process 800 ends. If next character is present, process 800 proceeds to step 830 to read the next character in the message string. A toggle bit is added in step 835. As the toggle bit is used to

indicate a next character, toggle bit will alternate between a "0" bit and "1" bit for each adjacent character in the string.

In step 840, an overlay is created for the next character read. The overlay is then appended to overlay string in memory in step 845. Process 800 then repeats from step 825.

Figure 9 illustrates a flow diagram of process 900 performed by the system in computer system 150, tablet 160 and mobile device 170 in accordance with embodiments of this invention. Process 900 is an embodiment of a process performed after the performance of process 800. Process 900 begins in step 905 by reading a first overlay in the overlay string. An image is captured by image capturing device 120 in step 910. The first overlay is then superimposed onto the image captured in step 915 to form a video message, such as the video message shown in figure 2c. The video message is then transmitted to virtual camera driver in step 520 of process 500.

In step 920, process 900 determines if there is another overlay in overlay string. If all overlays in overlay string have been read, process 900 ends. If all overlays in overlay string have not been read, process 900 proceeds to step 925 to read the next overlay in overlay string. Another image is captured by the image capturing device 120 in step 930. One skilled in the art will recognise that the same image captured in step 910 may be used since the image captured would likely be the same as that captured at step 930. Hence, step 930 is optional. The next overlay is then superimposed onto the captured image in step 935 to form the next video message to be transmitted to virtual camera driver in step 520 of process 500. Process 900 then repeats from step 920.

Figure 10 illustrates a flow diagram of process 1000 performed by the system in computer system 110 in accordance with embodiments of this invention. Process 1000 is an embodiment of a process performing step 610 in process 600. Process

1000 begins in step 1005 by capturing an image of video content received from network 190. In step 1010, process 1000 extracts a specific portion of the image. Particularly, the specific portion is the portion that includes the overlay. The number of image bits is then determined in step 1015. Verification is performed in step 1020 to determine the veracity of video message. Particularly, since 16 bits are used, the number of image bits has to be equal to 16 in order for video message to be considered valid. If number of image bits is not equal to 16, process 1000 ends and/or discards video message. If number of image bits is determined to be 16, process 1000 proceeds to step 1025 and converts the image bits to binary bits. The binary bits are then stored in a binary bits string in memory. In particular, all green boxes are converted to "0" bits and all red boxes are converted to "1" bits.

Process 1000 then proceeds to step 1030 to determine if there is a change in the toggle bit. The toggle bit is the last image bit in the overlay. The monitoring of the toggle bit is conducted by capturing the video content receive from the network at certain intervals and determining if there has been a change in colour in the last image bit in the overlay. If a toggle of the toggle bit is detected in step 1035, process 1000 repeats from step 1005. If a toggle of the toggle bit is not detected after a certain amount of time, process 1000 ends.

Figure 11 illustrates a flow diagram of process 1100 performed by the system in computer system 110 in accordance with embodiments of this invention. Process 1100 is an embodiment of a process after performing process 1000. Process 1100 begins in step 1105 to read the binary bits string stored in memory. In step 1110, 16 bits are extracted from the binary bits string. A character is determined and stored in a character string based on the first 8 bits of the 16 bits in step 1115. Process 1120 then determines if there is remaining unread bit in binary bits string. If there is any unread bit in binary bits string, process 1100 repeats from step 1110 to obtain the next character. Otherwise, process 1100 proceeds to step 1125.

In step 1125, process 1100 determines the first character in character string and remove the first character from the character string if the first character is the Start character, "~". In step 1130, process 1100 determines the last character in character string and removes the last character from the character string if the last character is the End character, "^". Process 1100 ends after step 1130.

Figure 12 illustrates a flow diagram of process 1200 performed by the system in computer system 110 in accordance with embodiments of this invention. Process 1200 is an embodiment of a process for performing step 615 in process 600. Process 1200 begins in step 1205 to read character string. A command associated with the character string is determined in step 1210. A command is generated and transmitted to controller 130 to control the consumer device in step 1215. After step 1215, process 1200 ends.

The above is a description of embodiments of a system in accordance with the invention as set forth below. It is envisioned that those skilled in the art can and will design alternative embodiments of this invention based upon this invention that infringe on this invention as set forth in the following claims.

Claims

1. A method for communication using a video conferencing application comprising:

receiving a command from an input from user;

generating an encoded command for said received command;

superimposing said encoded command onto a specific portion of an image received from an image capturing device to form a video message; and

transmitting said video message to a selected virtual driver, wherein said virtual driver transmits said video message to said video conferencing application in response to video message received by said virtual driver.

2. The method according to claim 1 further comprising:

receiving a video message;

monitoring for a change in said specific portion in said video message;

deciphering a command from said video message in response to a detected change in said specific portion of said video message;

generating a command message in response to deciphering said command; and

transmitting said command message to a controller for controlling a device.

3. The method according to claim 1 wherein said step of generating said encoded command comprises:

reading said command received;

encoding said command to 8-bit binary code to form a message string; and

storing said message string in said memory as said encoded command.

4. The method according to claim 3 wherein said step of receiving said command from said input from user further comprises:

inserting a Start character into said message string; and

inserting an End character into said message string.

5. The method according to claim 4 wherein said Start character is inserted to a beginning of said message string and said End character is inserted to an end of said message string.

6. The method according to claim 5 wherein said Start character is ~.

7. The method according to claim 5 wherein said End character is ^.

8. The method according to claim 3 wherein said step of superimposing said encoded command comprises:

reading a first 8 bits in said message string;

inserting said first 8 bits into an overlay;

adding 7 random bits to said overlay;

adding a toggle bit to said overlay; and

storing said overlay in said memory as an overlay string.

9. The method according to claim 8 wherein said step of superimposing said encoded command further comprises:

reading a next 8 bits in said message string;

inserting said next 8 bits into a next overlay;

adding 7 random bits into said next overlay;

adding a toggle bit into said next overlay; and

appending said next overlay to end of said overlay string in said memory.

10. The method according to claim 9 wherein said step of creating said overlay comprises:

creating a box for each bit in said 8 bits, said 7 random bits and said toggle bit in said overlay string;

determining each said bit that is a "0" bit in said overlay string;

filling said box of each said bit determined to be a "0" bit with a first predetermined colour;

determining each said bit that is a"1" bit in said overlay string; and

filling said box of each said bit determined to be a "1" bit with a second predetermined colour.

11. The method according to claim 10 wherein said step of creating said overlay further comprises:

generating a background in a third predetermined colour for said overlay; and

superimposing said boxes of said 8 bits, said 7 random bits, and said toggle bit onto said background to create said overlay.

12. The method according to claim 11 wherein said first predetermined colour is green

13. The method according to claim 12 wherein said second predetermined colour is red

14. The method according to claim 13 wherein said background is black in colour.

15. The method according to claim 9 wherein said step of superimposing said command onto said specific portion of said image further comprises:

reading a first overlay in said overlay string in said memory;

retrieving an image captured by an image capturing device; and

superimposing said first overlay onto said specific portion of said image captured by said image capturing device to form said video message.

16. The method according to claim 15 wherein said step of superimposing said command onto said specific portion of said image further comprises:

reading a next overlay in said overlay string in said memory;

retrieving a next image captured by said image capturing device; and

superimposing said next overlay onto said specific portion of said next image captured by said image capturing device to form said video message; and

repeating said step of reading said next overlay and retrieving said next image, said superimposing said next overlay onto said next image for the rest of overlays in said overlay string in said memory.

17. The method according to claim 2 wherein said step of monitoring comprises:

capturing an image of video content received from said video conferencing application;

extracting said specific portion from said captured image;

determining a change in a toggle bit in said specific portion of said captured image; and

repeating said step of capturing an image of video content, extracting said specific portion and determining said change in said toggle bit in response to a subsequent change in said toggle bit.

18. The method according to claim 17 wherein said step of determining said change in a toggle bit in said specific portion of said captured image further comprises:

determining a number of image bits in said specific portion;

comparing said number of image bits with a predetermined number; and

storing said image bits to form an image bit string in said memory in response to said number of image bits being identical to said predetermined number.

19. The method according to claim 18 wherein said step of receiving said video message in response to change in image further comprises:

discarding said specific portion from said memory in response to said number of image bits is not identical to said predetermined number.

20. The method according to claim 19 wherein said predetermined number is 16.

21. The method according to claim 18 further comprising:

reading said image bit string; and

generating a binary bits string based on said image bit string, wherein image bits in a first predetermined colour are assigned as bit 0 and image bits in a second predetermined colour are assigned as bit 1.

22. The method according to claim 21 wherein said first predetermined colour is green

23. The method according to claim 21 wherein said second predetermined colour is red.

24. The method according to claim 21 wherein said step of receiving said video message in response to change in said specific portion in said video image comprises:

reading binary bits string in said memory;

extracting a first 16 bits from said binary bits string;

determining a character based on a first 8 bits of said first 16 bits; and

storing said character to a character string in said memory.

25. The method according to claim 24 wherein said step of receiving said video message in response to change in said specific portion in said video image further comprises:

extracting a next 16 bits from said binary bits string;

determining a character based on a first 8 bits of said next 16 bits;

appending said character to said character string in said memory; and

repeating said step of extracting a next 16 bits, determining a character based on a first 8 bits of said next 16 bits, and appending said character to said character string in said memory for the rest of said binary bits string.

26. The method according to claim 25 wherein said step of receiving said video message in response to change in said specific portion in said video image further comprises:

reading a first character in said character string in said memory;

determining said first character is a predetermined Start character; and

removing said first character from said character string in response to a determination that said first character is said predetermined Start character.

27. The method according to claim 26 wherein said step of receiving said video message in response to change in said specific portion in said video image further comprises:

reading a last character in said character string in said memory;

determining said last character is a predetermined End character; and

removing said last character from said character string in response to a determination that said last character is said predetermined End character.

28. The method according to claim 27 wherein said Start character is ~ and said End character is ^.

29. The method according to claim 27 wherein said step of deciphering said command from said video message comprises:

reading said character string from said memory;

determining a command in association to said character string; and

generating a command message in association to said determined command.

30. A system for communication using a video conferencing application comprising:

circuitry a first processing unit configured to receive a command from an input from user;

circuitry said first processing unit configured to generate an encoded command for said received command;

circuitry said first processing unit configured to superimpose said encoded command onto a specific portion of an image received from an image capturing device to form a video message; and

circuitry said first processing unit configured to transmit said video message to a selected virtual driver, wherein said virtual driver transmits said video message to said video conferencing application in response to video message received by said virtual driver.

31. The system according to claim 30 further comprising:

circuitry a second processing unit configured to receive a video message;

circuitry said second processing unit configured to monitor for a change in said specific portion in said video message;

circuitry said second processing unit configured to decipher a command from said video message in response to a detected change in said specific portion of said video message;

circuitry said second processing unit configured to generate a command message in response to deciphering said command; and

circuitry said second processing unit configured to transmit said command message to a controller for controlling a device.

32. The system according to claim 30 wherein said step of generating said encoded command comprises:

circuitry said first processing unit configured to read said command received;

circuitry said first processing unit configured to encode said command to 8-bit binary code to form a message string; and

circuitry said first processing unit configured to store said message string in said memory as said encoded command.

33. The system according to claim 32 wherein said step of receiving said command from said input from user further comprises:

circuitry said first processing unit configured to insert a Start character into said message string; and

circuitry said first processing unit configured to insert an End character into said message string.

34. The system according to claim 33 wherein said Start character is inserted to a beginning of said message string and said End character is inserted to an end of said message string.

35. The system according to claim 34 wherein said Start character is ~.

36. The system according to claim 34 wherein said End character is ^.

37. The system according to claim 32 wherein said step of superimposing said encoded command comprises:

circuitry said first processing unit configured to read a first 8 bits in said message string;

circuitry said first processing unit configured to insert said first 8 bits into an overlay;

circuitry said first processing unit configured to add 7 random bits to said overlay;

circuitry said first processing unit configured to add a toggle bit to said overlay; and

circuitry said first processing unit configured to store said overlay in said memory as an overlay string.

38. The system according to claim 37 wherein said step of superimposing said encoded command further comprises:

circuitry said first processing unit configured to read a next 8 bits in said message string;

circuitry said first processing unit configured to insert said next 8 bits into a next overlay;

circuitry said first processing unit configured to add 7 random bits into said next overlay;

circuitry said first processing unit configured to add a toggle bit into said next overlay; and

circuitry said first processing unit configured to append said next overlay to end of said overlay string in said memory.

39. The system according to claim 38 wherein said step of creating said overlay comprises:

circuitry said first processing unit configured to create a box for each bit in said 8 bits, said 7 random bits and said toggle bit in said overlay string;

circuitry said first processing unit configured to determine each said bit that is a "0" bit in said overlay string;

circuitry said first processing unit configured to fill said box of each said bit determined to be a "0" bit with a first predetermined colour;

circuitry said first processing unit configured to determine each said bit that is a"1" bit in said overlay string; and

circuitry said first processing unit configured to fill said box of each said bit determined to be a "1" bit with a second predetermined colour.

40. The system according to claim 39 wherein said step of creating said overlay further comprises:

circuitry said first processing unit configured to generate a background in a third predetermined colour for said overlay; and

circuitry said first processing unit configured to superimpose said boxes of said 8 bits, said 7 random bits, and said toggle bit onto said background to create said overlay.

41. The system according to claim 40 wherein said first predetermined colour is green

42. The system according to claim 41 wherein said second predetermined colour is red

43. The system according to claim 42 wherein said background is black in colour.

44. The system according to claim 38 wherein said step of superimposing said command onto said specific portion of said image further comprises:

circuitry said first processing unit configured to read a first overlay in said overlay string in said memory;

circuitry said first processing unit configured to retrieve an image captured by an image capturing device; and

circuitry said first processing unit configured to superimpose said first overlay onto said specific portion of said image captured by said image capturing device to form said video message.

45. The system according to claim 44 wherein said step of superimposing said command onto said specific portion of said image further comprises:

circuitry said first processing unit configured to circuitry said first processing unit configured to read a next overlay in said overlay string in said memory;

circuitry said first processing unit configured to retrieve a next image captured by said image capturing device;

circuitry said first processing unit configured to superimpose said next overlay onto said specific portion of said next image captured by said image capturing device to form said video message; and

circuitry said first processing unit configured to repeat said step of reading said next overlay and retrieving said next image, said superimposing said next overlay onto said next image for the rest of overlays in said overlay string in said memory.

46. The system according to claim 31 wherein said step of monitoring comprises:

circuitry said second processing unit configured to capture an image of video content received from said video conferencing application;

circuitry said second processing unit configured to extract said specific portion from said captured image;

circuitry said second processing unit configured to determine a change in a toggle bit in said specific portion of said captured image; and

circuitry said second processing unit configured to repeat said step of capturing an image of video content, extracting said specific portion and determining said change in said toggle bit in response to a subsequent change in said toggle bit.

47. The system according to claim 46 wherein said step of determining said change in a toggle bit in said specific portion of said captured image further comprises:

circuitry said second processing unit configured to determine a number of image bits in said specific portion;

circuitry said second processing unit configured to compare said number of image bits with a predetermined number; and

circuitry said second processing unit configured to store said image bits to form an image bit string in said memory in response to said number of image bits being identical to said predetermined number.

48. The system according to claim 47 wherein said step of receiving said video message in response to change in image further comprises:

circuitry said second processing unit configured to discard said specific portion from said memory in response to said number of image bits is not identical to said predetermined number.

49. The system according to claim 48 wherein said predetermined number is 16.

50. The system according to claim 47 further comprising:

circuitry said second processing unit configured to read said image bit string; and

circuitry said second processing unit configured to generate a binary bits string based on said image bit string, wherein image bits in a first predetermined colour are assigned as bit 0 and image bits in a second predetermined colour are assigned as bit 1.

51. The system according to claim 50 wherein said first predetermined colour is green

52. The system according to claim 50 wherein said second predetermined colour is red.

53. The system according to claim 50 wherein said step of receiving said video message in response to change in said specific portion in said video image comprises:

circuitry said second processing unit configured to read binary bits string in said memory;

circuitry said second processing unit configured to extract a first 16 bits from said binary bits string;

circuitry said second processing unit configured to determine a character based on a first 8 bits of said first 16 bits; and

circuitry said second processing unit configured to store said character to a character string in said memory.

54. The system according to claim 53 wherein said step of receiving said video message in response to change in said specific portion in said video image further comprises:

circuitry said second processing unit configured to extract a next 16 bits from said binary bits string;

circuitry said second processing unit configured to determine a character based on a first 8 bits of said next 16 bits;

circuitry said second processing unit configured to append said character to said character string in said memory; and

circuitry said second processing unit configured to repeat said step of extracting a next 16 bits, determining a character based on a first 8 bits of said next 16 bits, and appending said character to said character string in said memory for the rest of said binary bits string.

55. The system according to claim 54 wherein said step of receiving said video message in response to change in said specific portion in said video image further comprises:

circuitry said second processing unit configured to read a first character in said character string in said memory;

circuitry said second processing unit configured to determine said first character is a predetermined Start character; and

circuitry said second processing unit configured to remove said first character from said character string in response to a determination that said first character is said predetermined Start character.

56. The system according to claim 55 wherein said step of receiving said video message in response to change in said specific portion in said video image further comprises:

circuitry said second processing unit configured to read a last character in said character string in said memory;

circuitry said second processing unit configured to determine said last character is a predetermined End character; and

circuitry said second processing unit configured to remove said last character from said character string in response to a determination that said last character is said predetermined End character.

57. The system according to claim 56 wherein said Start character is ~ and said End character is ^.

58. The system according to claim 56 wherein said step of deciphering said command from said video message comprises:

circuitry said second processing unit configured to read said character string from said memory;

circuitry said second processing unit configured to determine a command in association to said character string; and

circuitry said second processing unit configured to generate a command message in association to said determined command.