Dynamic Resource Management using Gesture-Based User Interface

Sripradha K, Nikhil Kamath K, Nilesh Nayak H, Ram P Rustagi
Department of Information Science and Engineering
PES Institute of Technology, Bangalore, India
Email: www.pes.edu

Abstract—We propose and implement a user-convenient method for dynamically utilizing resources across varying platform devices using air gesture-based user interface. Resource-sharing is the most ubiquitous task found in modern days. Most commonly used methods include usage of bluetooth, pen drives, external hard disks, emails, chat applications and social networking websites for sharing images, video/audio resources, url and other contents. The proposed idea eliminates the elaborate process of choosing the resources to be shared and also selecting the method of transfer, be it any of the ways mentioned above. Our idea encapsulates the entire process into a single gesture for the resource-sharing mechanism. In this work we put forward a user friendly, intuitive approach for sharing of resources among cross-platform computing devices using ip network as a medium of transfer and human hand gestures as user interface conjugated with dynamic detection and rendering of the resource.

I. INTRODUCTION AND RELATED WORK

With the advancement of technology, machines from big computers to small gadgets have gained lot of importance in humans’ life. These have not only made life simple but also made it more comfortable and luxurious. The idea of easy interaction with these machines for the sake of user convenience is the key behind its success. Resources form a major part of a computing environment. Creation and sharing of documents, media files and other similar resources are the most ubiquitous actions performed by users. Any method that can make the whole process convenient to the user will be a boon to the society.

Human computer interaction (HCI) [1] aims to improve the interactions between users and computers by making computers more user friendly and receptive to one’s needs. The method of interacting with machines has travelled a long way with the design of new technologies. In recent years, there has been a great deal of interest towards methods on computational vision due to its ability to recognise human gestures in a natural way [8].

Over the years, lot of work has been done with respect to hand detection technique. Most of the techniques in general deal with the study of pixel color and intensity of image [4] rather than features of the hand. To simplify this process, many gesture recognition applications adopted the use of uniquely coloured gloves or markers on hands or fingers [9]. In addition, using a controlled background makes it possible to localize the hand efficiently and in real-time [10]. In Gesture-based Computer Control System applied to the Interactive Whiteboard [6], a gesture-based computer control coupled with the dedicated touchless interactive whiteboard is presented. This depicts how a projector paired with camera can be used to interact with the digital world such as zooming the contents, controlling the mouse using hand gestures. In today’s world, many systems allow user to control devices and its application with one or more gestures to ease the interaction. Flutter [7] is a gesture recognition application that allows users to control media playing applications such as iTunes, Spotify, VLC and QuickTime using hand gestures. The application utilizes gesture recognition technology that works with the webcam on a user’s computer.

The resource sharing today mostly is driven by human interaction. Our work enhances this by making use of machine to machine communication. Internet of Things(IOT) [2] is one such concept in which physical objects communicate through internet. It represents every object as a digital entity which enables its usage from anywhere, anytime. The growing global Internet of Things concept has evolved from convergence of wireless technologies, micro-electromechanical systems and the Internet. According to Gartner [11], there will be nearly 26 billion devices connected on Internet by 2020. An example application of IOT combined with sensor technology for automobiles, is that a car would know when to hit the brakes before a driver does. The possibilities of where IOT is applicable are endless. Hence, with the recent progress and development in the field of IOT and related technologies, together with HCI, the two domains could support the need for evolution of new and innovative ways. This paper focuses on a novel method to combine the domains of HCI and IOT and has articulated an elegant and innovative method to share resources among users, by making the resource-sharing mechanism intuitive and convenient to the users of the application.

Consider a classroom scenario where the teacher wants to share a resource with students that she/he is currently using. In cases such as these, conventionally pen drives, emails, bluetooth or social networking sites are utilized. Lately, storing data on cloud which can be shared without duplication is gaining importance. Syncing of data to a common storage space is also of high interest. Devices such as Apple TV, Google TV and applications including AirDrop facilitates sharing of resources across devices running on same platform. All these conventional methods involve additional hardware, or limit the sharing to few devices; also engaging the user in choosing the resource to be shared from the file sys-
tem/database, travelling to the destination storage space to retrieve it and also possess the external devices. Usage of mails or social networking websites mainly requires sender to login, attach and send to the appropriate receiver which should be downloaded to local system for viewing. Web conference is also an intelligent way of sharing resources. It is a service that allows meeting events to be shared anywhere with the help of internet. A lot of advancement is done on this by WebEx [15], which is a Telecommunications software and services industry that provides applications related to on-demand collaboration, online meeting, web conferencing and videoconferencing. WebEx web conferencing [16] allows a person to meet anyone, anywhere, in real time. Further, WebEx provides a method for everyone in the conference to see the same thing while one talks by combining desktop sharing through a web browser with phone conferencing and video.

The user interface of the existing systems involves usage of keyboards, mouse, touch pads, etc. With the advent of Microsoft's kinect, leap motion devices and applications like Flutter which uses camera, air gestures have made their way into the consumer market. And so, we addressed a need by providing a cutting-edge solution, which is simple and easy to use by constituting the following.

1) Provide an air gesture based user interface among communicating devices facilitating the resource-sharing process without any additional hardware or sensors. We used human hand-detection mechanism for defining and recognizing gestures and these gestures were mapped to appropriate actions in the back-end. The complete details of the implementation are explained in section 2.

2) In order to render resources on recipient’s device, the presence of the device must be known to the sender. Hence we have developed a protocol for discovering all the active devices within a network qualified for sharing. We provide the details in section 3.

3) Our work eliminates the constraint of manually navigating to the resource destination and provides an elegant solution for obtaining the resource which is currently rendered in the user’s active window. The process is explained in detail in section 4.

4) Once the resource and its details are recognized, it is shared with the intended receivers. Section 4 elaborates the ways in which users can choose the intended receivers.

5) Conventionally, after sharing the resource, it is saved on the receiver’s end in the file-system/database. The user is expected to open the resource on the destination device via a default application to access it. Our work simplifies the process and makes it more convenient to the user not only by saving the resource as a local copy on the receiver’s end, but also by rendering the received resource as an active application on the receiving end. Further explanation is provided in section 4.

6) Users are provided with options for setting the identity, mode of transfer and also facility to form groups to which he intends to share resource. These options are explained in details in section 5.

II. HAN D GESTURE RECOGNITION AND DEFINITION

Gesture recognition is a process that enables users to communicate with devices in a humanly and natural way. Initially, Motion Detection through Background Subtraction technique was used during the trivial stage of the work. Using this technique, the difference in frames was determined and motion was detected. This can be used to get the shape of object partially in motion. However, it suffered from few disadvantages and mainly cannot detect object in its static position. To overcome those disadvantages, Disparity Mapping technique was used. Disparity refers to the variation in location of an object in correspondence to two images, as seen by the left and right eye, which is created due to parallax of special type of cameras known as stereo cameras. This technique is very useful to detect the object depth, like hand which will be closest to camera during gestures. A very useful and highly accurate and may be the next big thing in Computer-Vision with cameras being replaced with kinects and other such powerful devices. However this technique is not possible with the web-camera presently built into laptops and usage of special cameras entails additional hardware cost. Detection of gestures is also possible by drawing contours around hand or by using gloves, colored bands etc. But this method is accurate when the background is monochrome and stagnant. The primary step in our process is to detect human hand and in this work, with our research on the above mentioned methods, Haar Cascade Classifier[5] was chosen for the detection. The advantage of the this technique is that we could train the machine to differentiate the objects of interest from the rest. Moreover, for real-time detection of the object, it would be a constant time measure. Haar classifier algorithm is very efficient in terms of computing power though the initial training requires large amount of resources. Once the hand is
detected, the next step in the work maps the hand movement to define gestures. These gestures constitute the input that is mapped to a particular action in the back-end.

OpenCV [14], an open source library, is used to support real time computer vision and thus it serves the purpose for gesture recognition. The cascade classifier training method of OpenCV was used to train the machine with a set of positive images, which represents the region of interest (ROI) and negative images, which represents all other images without ROI. For this purpose, a training data set of 100 positive images and 2000 negative images were used to create 3500 sample images by applying transformations and distortions. These samples were fed into the train cascade tool of OpenCV to obtain a trained classifier. This is then used to detect hand in real time scenario comparing with the frames captured from web camera. The entire process of gesture recognition is depicted in Fig 1.

The gestures detected are of 2 types based on the mode of transfer. The two modes available are,

1) Send to a predefined group or to all.
2) Peer to peer.

The functionalities of the two modes are addressed in section 4. A general gesture called “Send-all” uses one gesture which can be used in both the modes. In this case the gesture is a closed fist followed by open palm as shown in Fig [2]. In the first mode, only one gesture is used and it is defined as open palm moving from left to right or from right to left as shown in Fig [3]. In the second mode, two gestures are used. One on the sender side and the other on the receivers end. The gesture again is open palm moving from left to right or from right to left on both ends. The gesture at sender initiates the transfer and the gesture at receiver end completes the transfer and rendering of the resource. With the detection of an accurate gesture, properties of active window currently accessed by the user or details of the foreground application are examined. A typical use case of the above implementation is when a user copies a URL from a key requirement before initiating a resource-sharing process. Hence an application layer protocol is developed on top of multicast technique to spot active devices. This protocol enables devices/users to join a multicast group and share standardized messages with other active devices in the network. A non-periodic increasing exponential time-out is maintained which facilitates early detection of other devices and also to announce the presence of user’s device in the network. The information about the active devices is maintained in a local data structure. The data structure holds the user identity, IP address and the MAC address of all the active devices. This data structure is then used by the resource-sharing mechanism for identifying intended receivers. The overall working of the protocol is shown in Fig 4.

IV. RESOURCE-SHARING MECHANISM

The next important part after gesture recognition and device discovery components is the dynamic detection of active resource and sharing it based on the mode and the interpreted gesture. Once a valid gesture is recognized, it is mapped to one of the three actions stated below.

1) Share resource with predefined groups initiated by setting mode to 1.
2) Share resource with an intended peer initiated by setting mode to 2.
3) Share resource with all the active users in the network initiated by “Send-all” gesture.

With the detection of an accurate gesture, properties of active window currently accessed by the user or details of the foreground application are examined. A typical use case of the above implementation is when a user copies a URL from

III. DEVICE DISCOVERY PROTOCOL

Detection of active devices within the network and forming a group of devices among which the resource is to be shared is

The gestures detected are of 2 types based on the mode of transfer. The two modes available are,

1) Send to a predefined group or to all.
2) Peer to peer.

The functionalities of the two modes are addressed in section 4. A general gesture called “Send-all” uses one gesture which can be used in both the modes. In this case the gesture is a closed fist followed by open palm as shown in Fig [2]. In the first mode, only one gesture is used and it is defined as open palm moving from left to right or from right to left as shown in Fig [3]. In the second mode, two gestures are used. One on the sender side and the other on the receivers end. The gesture again is open palm moving from left to right or from right to left on both ends. The gesture at sender initiates the transfer and the gesture at receiver end completes the transfer and rendering of the resource. With the detection of an accurate gesture, properties of active window currently accessed by the user or details of the foreground application are examined. A typical use case of the above implementation is when a user copies a URL from
the browser and pastes the same in a chat window or send the same to a recipient user via email. One has to open the browser and do mouse clicks/drags to select the URL. Our work completes the process automatically by simulating keyboard key press by initially getting the web browser details. Similarly for documents/media files, a general method employed is to traverse to the destination of the resource and then send the resource by means of copying it to flash drives or email attachments. In this case, from the foreground application the resource is identified and all the related meta data such as resource name, location, size, type are extracted. Once all the details are obtained, the resource transfer process is initiated using IP network as a medium of transfer. Hence the resource to be shared must be in the foreground. The intended users are determined based on the gesture recognized and also on the mode of transfer set by the user. Thus any resource that is being currently viewed by the user can be shared with intended receivers easily with just one gesture.

The receiver receives the shared resource if she/he is running the application on the required device. Meta-data contents are first examined and based on the type of the resource, be it a documents/media resources or URL, an appropriate default application is launched on the recipient system. If the type of the resource is a URL, the default web browser is launched and the contents of the webpage are rendered. If the type of the resource is a document, text, audio/video or other contents, the resource is rendered using the corresponding default application as set on the receiver's device for viewing. Also, a local copy of the resource is saved in the file system/database on the receiving device for future use. The demonstration of resource-sharing mechanism can be found in the URL https://vimeo.com/96798449. The design details of the sharing system is represented in Fig 5 and Fig 6.

V. Configuration Option

The user can customize and finetune the resource sharing process by identifying himself with an identity. The choice of the mode of transfer can also be set by the user. A “Scan devices” option is provided which when clicked lists all the active users in the network. Then a set of users can be grouped by selecting from the list provided. This user group is saved and is put to use when the type of transfer is set to mode 1 as defined previously. The users are identified persistently using MAC addresses of the systems. A snapshot of the configuration is seen in Fig 7. The user interface that appears on click of “Scan devices” is shown in Fig 8.

VI. Implementation and Testing

Making use of Haar Training method has been a daunting challenge throughout the period of our work and it presented quite a few challenges with respect to performance. Training a highly accurate classifier takes a lot of time, a huge number of samples and heavy computing power. As stated earlier, a training data set of 100 positive images and 2000 negative images were used to create 3500 sample images. The process of training machine depends on a variety of factors, including the quality of the images, the object that needs to be recognized, the method to generate the samples, the CPU power and other related conditions. Once the pictures of object were collected, it had to be cropped so that only the desired object is visible. The CPU characteristics that was employed for the
training process were 32 GB RAM and 8 Core processors working up to almost 100% efficiency of all 8 cores. The time taken to train for above mentioned numbers is 60 days. The output is an XML-generated file which is used against real-time frames from camera, and detection of the object for gesture recognition.

The gesture recognition system is trained to work most of the times in good environment conditions. It still needs to be trained further with respect to the cascade classifier generated in the early stages of the work. The various issues faced are with respect to different lighting conditions and environments. The system detects too much of false positives in real-time capture scenario with respect to the initial trained classifier. It is expected that with more training, better accuracy results could be attainable with more positive and negative images collected and fed for training.

The method supports different types of media files like MP3, MP4, wmv and other such formats for sharing and rendering. A major issue is concerning the size of such media files. The work has been tested for resources/files with sizes ranging from small MP3 having size of 2-7 MB to large video/movie files having size of 500-700 MB. When compared to the conventional means of transfer of such files through flash drives or external hard disk, it is quite acceptable to share files through this method via IP network. The reasons being that Flash drives or external Hard disk methodology of transfer of files requires

1) Compatibility with respect to different Operating systems.
2) Device driver detection for every new device.
3) Could crash during data transfer, if unplugged without noticing.

After having observed that large media files take quite a reasonable amount of time for transfer through network due to various issues like bandwidth which are against certain needs of the users, it does not prove to provide a full guarantee that this work serves all the purpose of the user. But it certainly does provide an indication that with the advent of technologies like IOT and HCI, the next world is sure to remain connected amongst computing devices and the transfer of such resources is mostly going to happen via applications having internet as the medium of transfer.

The Operating systems on which our work is currently used and tested includes Ubuntu 12.04 and above, MAC OS-Mavericks(10.9) and for hand held devices Android OS 4.1-JellyBean and above. For desktops, the application is tested on systems having 4 GB RAM and for hand held devices it runs on 512 MB RAM.

Testing for security has always been an issue when devices communicate with each other. For peer to peer mode, only when the gesture is done from the sender, after which the receiver acknowledges with a return gesture, the resource sharing is facilitated. This forms a practical implementation of implicit authentication by itself between the sender and receiver.

Device discovery protocol could pave way for many third party attacks. A wireshark dissector has been built for debugging purposes, where data transfer over the network is examined and tested between computing devices. Currently the application is implemented only for transfer of resources from desktop to desktop and mobile devices and is yet to be extended for transfers from mobile devices to desktops and similar devices.

VII. CONCLUSION AND FUTURE WORK

In this paper an attempt has been made to automate the elaborate process of sharing resources making it naturally convenient to the users. The unique feature of one gesture and rendering the resource enumerates the convenience. Also the usage of hardware interaction such as keyboards, mouse is immensely reduced by providing an intuitive and ingenious user interface powered by hand gestures. We believe that our approach can be useful in everyday scenarios such as a classroom where the teacher would like to share the study materials with students or when a document being read on desktop has to be transferred to a handheld device for portability or when a link has to be shared with a bunch of friends or project mates for reference related purposes while carrying out the project. All these activities can be done by just a gesture, thus increasing the usability of the application and reducing the conventional methods mentioned in the abstract section above. The paper thus focuses on effective collaboration among devices and various existing technologies, mainly HCI with gesture recognition and IOT to ease the daily work and also enhancing the conventional methods that are being used. The method that is explained in this paper requires no additional
we call hardware thus making it free from any additional hardware cost and suitable for the users with the inbuilt camera. The scope for improvement can be seen in parts of the work. The gesture recognition system can be made more accurate if devices like kinect or depth aware cameras are used. But this adds additional cost to the application. As discussed previously, implicit authentication is observed in case of peer to peer, but better authentication mechanism can be added in future for sharing resources with groups or all the active users. Gestures can be defined to control the movement of mouse, choose files which is required to be shared and also to provide authentication. The work can be extended for defining more tasks such as moving palm upwards to sync resources to cloud, thumbs down to print the currently opened resource or download a particular resource or draw patterns in the air for authentication. The gestures not being restricted to hand gestures, can be audio and visual gestures such as a snap of fingers to power off the system or clap to initiate the resource transfer, retina recognition for authentication. Similarly, air gestures can be used to manage WebEx web conferences. Hence, an extension of the same concept for users not in local network could be made possible with the advent of cloud based sharing in future. Then discovery of device has to use other protocols and may not depend on multicast technique. Currently internet is used as a medium of transfer of resources. Bluetooth can act as an alternative here.

ACKNOWLEDGMENT

The authors would like to thank PES Institute of Technology for providing us with a congenial environment to carry out this work.

REFERENCES