Hand gesture based X-ray image controlling using Convolutional Neural Network

Rutika Mhatre[1], *Bhakti* Dhage[2],*Vishesh* Kwatra[3] ,and *Pallavi* Chavan[4]

[1]Department of Information Technology,Ramrao Adik Institute of Technology,Navi Mumbai,India rutikamhatre21@gmail.com

[2]Department of Information Technology,Ramrao Adik Institute of Technology,Navi Mumbai,India dhagebhakti0123@gmail.com

[3]Department of Information Technology,Ramrao Adik Institute of Technology,Navi Mumbai,India kwatra.vishesh@gmail.com

[4]Department of Information Technology,RAIT,D.Y. Patil Deemed to be University,Navi Mumbai,India pallavi.chavan@rait.ac.in

Abstract.This paper proposes a novel computer vision based system that allows doctors, surgeons and other physicians to control X-Ray images just by using simple gestures thus eliminating the need of traditional devices like mouse and keyboard. This will help reduce the risk of contamination in sterile environments like those found in the hospitals and it will also help in preventing the spread of covid by not allowing contact with contaminated surfaces.It is implemented using CNN model. CNN is specially used for image recognition as well as processing.The system detects gestures through in-built webcam and converts it into corresponding computer commands to perform its associated tasks.

Keywords Computer Vision, X-ray, Keyboards, Contamination.

1. Introduction

Computer information technology is continuously growing in the Hospital domain. It is necessary to handle these technologies in a safe manner to avoid serious mistakes, because this may lead to possible fatal incidents. Real-time imaging review is important in surgery, particularly during the operation. Currently, Doctors and Nurses use keyboards and mouse to handle the images but these are not the appropriate devices for this task. This may lead to the spreading of disease causing pathogens from one user to another.

Many studies have tried to solve this problem by using various devices like leap motion devices as well as the use of Microsoft Kinect Technology to enable contactless control of the computers allowing medical images to be controlled without the risk of bacterial contamination.

However the main issue with all of these approaches is that all of them require some sort of expensive hardware device which may or may not be available in some regions. Also, not all hospitals and clinics can afford to buy expensive hardware devices like kinect or the leap motion sensor. Often small clinics have limited budgets and they have to spend most of their budget in buying the essential things that are very much crucial to keep the clinic up and running. So our project is aiming to resolve this issue by enabling the identification of gestures just by using a common webcam that is found in-built in most computers and this will allow our system to be accessible to even the smallest of clinics.

Hand gestures are a kind of body language that can be conveyed through different actions, for example one can use their palm to move the cursor, use their index finger for clicking, etc. These gestures are again divided into two types i.e. static and dynamic gestures. Static hand gestures means stable hand gestures whereas the dynamic gesture consists of hand movements such as waving. One of the main advantages of using static hand gestures is that they are read more accurately and require less complex algorithms for their recognition making the system fast and resource efficient. That is why we have opted to use static gestures instead of dynamic gestures for faster gesture recognition and more accuracy.

We are using hand gestures for manipulation of x-ray images. The major advantage of sterility as gesture is basic form of communication and it is often used when people speak to each other . It can perform actions like moving images, zooming images,etc. Related work i.e. various literature surveys appear in section 2 and proposed system of the project specified in section 3 which include initial planning,training a CNN,mapping of gesture,debugging,etc.Section 4 contain the result of the project.Final conclusion of the project are provided in section 5.

2. Related Work

Reviewing real-time x-ray images during surgery is very important, especially during operations. Traditionally devices like mouse, keyboard and touchpads are used. But these devices are reliant on physical contact which can lead to the spreading of infection from one user to another. So we have developed a solution that will allow doctors to manipulate and view x-ray images without the usage of devices that require physical contact like keyboard and mice.

[1] In this paper, tests have been done to check for the presence of pathogenic microorganisms on keyboards and mice. A total of 35 samples were collected from different computer labs of an institute in Libya. The samples collected were cultured for further analysis. The results demonstrated that all 35 samples were contaminated with five pathogenic bacteria (E.coli supp. Salmonella spp. Klebsiella spp. Pseudomonas spp. and Staphylococcus spp.). All of the samples in this study were contaminated. This shows how common electronic devices like keyboard and mouse can act as a carrier for pathogenic microscopic organisms. Regular Cleaning of these devices minimizes the bacterial level necessary to diminish the bacterial level.

[2] In this paper, 100 keyboards were tested for bacterial contamination and in the results it was found out that ninety five of them were contaminated with some sort of bacteria. Streptococcus, Clostridium perfringens, Enterococcus (including one vancomycin-resistant Enterococcus), Staphylococcus aureus, fungi, and gram-negative organisms were solded. This really shows the importance of keeping computer equipment clean so that it does not lead to the spreading of disease causing pathogens from one user to another.

[3] During surgery, the surgeon usually is not the person who directly controls the computer. Instead the doctor instructs his/her assistant to perform operations on the computer. This type of communication is not efficient at all and it can be a source of frustration in the operating room if there is misunderstanding between the surgeon and the assistant. This paper describes the design of a joystick like device that can be used to control the computer without needing verbal communication with the assistant. The surgeon controls the computer by placing the probe into the bese and manipulating it like a joystick. The base provides force feedback and permits intuitive clicking with the probe.

[4] This paper proposes a touchless visualization system for computer aided surgery (CAS) which can be used to manipulate a patient's 3D anatomy model through the use of Microsoft kinect. Such anatomy models play a really important role during surgery as they help surgeons to get a good idea on how to go about the procedure and where the problem actually is within the patient. Microsoft Kinect was released alongside the Xbox 360 game console as an interactive way to play games where your body acts as a controller to provide inputs to the game. The motion tracking provided by the Kinect was very accurate and responsive. The depth and skeleton information from Kinect is used effectively to recognise the various gestures made by the user and translate them into actions to control various types of medical images.

[5] In this paper, a gestural interface based on the Leap Motion sensor device is developed and the gesture operations are integrated into PetaVision3D, and in-house 3D PACS software for clinical use. An optional toot pedal was also implemented so that the user could make gestures using just one hand. The main goal was to translate gestures into actions to control medical images. The Leap Motion Controller is an optical hand tracking module that captures the movement of your hands and allows you to interact directly with the digital content on Windows PCs. The Leap Motion sensor itself consists of two cameras and some infrared LEDs. It also has a wide angle lens to increase the interaction space so that gestures could be detected over a larger area.

[6] This paper is more of a general overview of the work that has been done in the field of hand gesture recognition and the comparison of various techniques that have been used for the same. The hand gesture recognition techniques have been compared on a variety of criterias like performance, algorithms used, drawbacks, type and number of gestures, the dataset used and so on and so forth. This paper also discusses the applications of hand gesture recognition in a variety of fields giving us a very clear iden on how useful hand gesture recognition can be in the future.

[7] This research paper aims at offering new possibilities for people to interact with machines using intuitive gestures that are then translated into computer commands. Their system is based on computer vision and it allows their system to recognise a number of static and dynamic gestures. Their system can recognise six static and eight dynamic hand gestures. There are three main steps involved when it comes to gesture recognition in their system : Recognising the shape of the hand, tracing the detected hand only if it is a dynamic gesture and then converting the acquired data into a computer command to perform an action. Their system shows an accuracy of around 93.09 %.

[8] This research paper cites that with the ever growing inclusion of computers into our society, the existing modes of interactions with computers (mouse and keyboard) can become a bottleneck hindering the information flow between the machines and humans. In order to avoid this bottleneck in the future and allow for effective exchange of information between humans and computers, Vision based Gesture recognition can be a powerful tool for achieving the naturalness

required for Human Computer Interaction (HCI). This paper also highlights the work that has been done in the field of computer-vision based analysis by many researchers and it emphasizes the importance of having such a system in place for the future.

[9] This paper highlights that the attractiveness of computer games can be enhanced by vision-based user inputs. A system like this should have a very quick response time (less than a video frame) and should not be costly so that it can be produced for the masses. These constraints were met by using special algorithms that were tailored to a particular hardware. A chip called the artificial retina chip was developed which allowed for fast image processing. The algorithms were then developed to use the capabilities of the chip to provide a very interactive response to player's hand or body movements at 10 msec frame time and at a low cost. These interactions were then demonstrated in several games.

[10] In this paper, research has been carried out for the recognition of dynamic hand gestures. The gestures that have been selected are a sequence of distinct hand poses which can undergo motion and different changes. A recognition engine was developed that was able to accurately detect these gestures despite individual variations. The engine also has additional capabilities like detecting the starting and ending of gesture sequences in an automated fashion. One of the main advantages of this system is that it is able to stay accurate despite the background clutter that might be present and it uses skin color for tracking and recognition. This was implemented on standard hardware which allowed for the recognition of dynamic hand gestures in real time.

[11] This paper uses a camera and two colored markers colored red and green that are worn on finger tips that are used to generate the desired hand gestures. For detecting the markers and tracking they have used template matching with Kalman Filter. When the gestures are detected the commands are issued to the system to perform the desired action. Their system can perform a variety of system actions like movement of the cursor, mouse button clicks and zooming. This system is feasible on devices where touch-screen is not an option like large screens or projected screens and it even works on desktops.

[12] This paper cites the importance of human computer interaction and how traditional mouse and keyboard can be a huge barrier and bottleneck when it comes to HCI. In this paper, a robust and marker-less hand gesture recognition has been proposed that allows their system to effectively track both static and dynamic gestures and perform various system tasks like opening websites, applications, moving through slides on microsoft powerpoint. Their overall system consists of two parts which includes the front end and the back end. They have used three hardware modules which include the detection module, the camera module and the interface module.

[13] This paper makes the use of sensible shape based features like COM (Center of mass), orientation, fingers that are folded, fingers that are raised and their position in the image. A web cam is used to get the image frames which are then pre-processed to remove any background noise. Then with the usage of K-means clustering, the hand is separated from the rest of the background in order to calculate the shape based features. A voice processor is also used that plays a voice through the speaker whenever a gesture is recognised that allows even blind users to use their system.

3. Proposed System

Hand gestures are a form of nonverbal communication that can be used in several fields such as communication between deaf - mute people, robot control, human-Computer interaction (HCI), home automation and medical applications. The system can operate and perform specific user tasks efficiently. A system that can accurately and quickly recognise gestures and perform tasks makes the interaction between a computer and a human very smooth and hassle free.

3.1 - Planning of our Project

Our proposed idea was executed in 5 major phases:

Phase 1 : Planning our Proposed System
Phase 2 : Determining the libraries and software required
Phase 3 : Using a dataset to train a CNN model
Phase 4 : Using the trained model to recognise gestures and mapping them to keys
Phase 5 : Cleaning the code and adding the final touches.

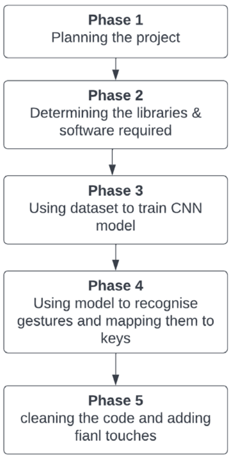
Phase 1
Planning the project
Phase 2
Determining the libraries & software required
Phase 3
Using dataset to train CNN model
Phase 4
Using model to recognise gestures and mapping them to keys
Phase 5
cleaning the code and adding fianl touches

Figure 1 - Planning of our project

3.1.1 - Initial Planning

In the first phase, we planned our project in order to achieve our desired goal. This phase mostly involved reconnaissance and surveying to determine the extent to which our project can help and make the lives of medical professionals easier. We also did literature surveys and studied a number of research papers to determine the amount of work that has been done in this field and the technologies that already exist and used by medical professionals.

3.1.2 - Libraries

In the next phase we determined the libraries that we wish to use and the environment that we want to use for coding our project. To accomplish this, we studied and selected the libraries that are available for us to use for free of cost to make sure our system is as cost effective as possible. Once we selected the libraries that we require, we researched and analyzed these libraries so as to import only the packages which are essential for our use case so as to not slow down our system by importing any unnecessary packages. For our proposed system we have used a number of libraries, some of which are:

- Numpy
- Keras
- Tensorflow
- OpenCV
- PyAutoGUI

NumPy is a library for the Python programming language that adds support for large, multi-dimensional arrays and matrices, along with a number of comprehensive mathematical functions and linear algebra routines to operate on these arrays. NumPy arrays are way faster than Python lists for performing large numerical operations as NumPy array functions are evaluated internally in C++ which makes their runtime way faster. This library is open-source and its high level syntax makes it very easy to understand for programmers of any skill level.

Keras is an open-source software library that uses Python and provides a way to interact with the TensorFlow library to create artificial neural networks and it is also used for other machine learning and artificial intelligence applications. It has a number of commonly used neural-network building blocks (layers, activation functions, optimizers) and a large number of tools that allow for the easy processing of a variety of data types and makes the coding process for writing a deep neural network easier. Keras is used by a large number of individuals and scientific organizations around the world including CERN, NASA and NIH

TensorFlow is an open-source software library developed by Google used for machine learning and artificial intelligence purposes. It can be used for a variety of purposes but it mainly focuses on the creation and training of deep neural networks. The Keras library runs on top of TensorFlow. It is written in C++, Python and CUDA. Having a GPU (Graphical Processing Unit) greatly speeds up the deep learning processing rather than just relying on the CPU for training. It accepts data in the form of arrays with multiple dimensions called tensors.

OpenCV is an open-source software library that allows for real-time computer vision. It can be used in conjecture with other libraries like Keras and TensorFlow to build complex neural networks and other machine learning applications. It is written in C++ but it also has a Python API. By using this library, it is possible to process images and videos for a variety of applications such as street view image stitching, medical image analysis, interactive art installations etc. It also supports GPU acceleration that allows it to use the CUDA cores for faster processing of data.

PyAutoGUI is a python library that allows for control of mouse and keyboard strokes through python scripts. It can be used for automating interactions with applications, taking screenshots, displaying messages, interacting with application windows (moving, resizing, minimizing, maximizing), moving the mouse cursor to a specific position on the screen etc. It can also be used to press multiple keys at once which are called macros. This can be very useful especially in photo and video editing applications where there is a macro for every task.

3.1.3 - Convolutional Neural Network

Deep learning is a subset of machine learning and it basically tries to imitate the structure and functioning of a human brain and allows computers to think and perform tasks like humans and even allows them to “learn” things without being explicitly coded. A Convolutional Neural Network is a deep learning model that can be used to work with data that is far more complicated, unstructured and varied such as audios, images or text files. This kind of data cannot be used with traditional machine learning algorithms. A neural network has the following components :-

- Input Layer
- Hidden Layer
- Output Layer

The **input layer** is the layer from which the data is fed as input to the neural network. The data can be anything ranging from text files to audio or images.

The **hidden layer** is responsible for the processing of data and feature extraction by performing complex computations. This layer/layers has weights and biases that are continuously updated as a part of the training process. After the computation is complete, the output is passed on to the output layer.

The **output layer** is responsible for predicting the values by using suitable activation functions. The output can be either numerical or categorical in nature.

Keras offers two ways to build neural networks :-

1. Sequential
2. Functional

The **Sequential Model** is the easiest way to build a model in Keras. It allows the user to build a model layer by layer and the data flows from one layer to another until it reaches the final layer. But the drawback of this model is that the sharing or branching of layers is not allowed. Also, it is not possible to have multiple inputs and outputs.

The **Functional Model** is more flexible than the sequential model as it allows for branching and sharing of layers and enables you to create more complex models. In fact, you can connect layers to any other layer. This model also allows for multiple inputs and outputs.

3.1.4 - Training a CNN

In the next phase of our project, we trained a sequential CNN model to recognise various gestures that the user makes in front of the webcam. In order to train our model, we have made use of various datasets available to us from Kaggle. These datasets had a large number of images for each gesture which we have used to train our model. Before training our model these images had to be pre-processed so as to make the training process as seamless and as accurate as possible. These datasets are imported and the CNN model is trained until a good accuracy rate is achieved. Our trained CNN model is capable of recognising 5 static hand gestures :-

1. Closed Palm
2. Open Palm
3. Point Left
4. Point Right
5. Point Up

3.1.5 - Mapping of Gestures

In the next phase, we mapped the hand gestures to specific key(s) or keyboard strokes which are used to control the X-Ray images within the software. In order to accomplish this we mainly made use of the PyAutoGUI library. The user needs to have a webcam for inputting the gestures. The webcam is used to gather video frames and each frame is then run through our CNN model to determine what gesture is the user doing and our system accordingly performs the tasks within the software to control the X-Ray images.

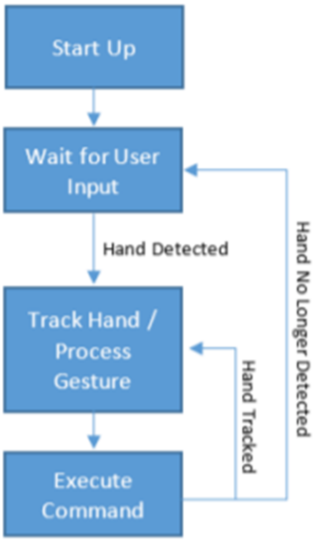

Figure 2 - Block Diagram of our System

3.1.6 - Debugging

The final stage of our project involved adding final touches to the code and documenting it so that it is easy to understand whenever any user tries to modify the code for their own usage. We also removed any unwanted lines of code and any unused libraries in order to further optimize our code.

3.2 - X-Ray Imaging

X-Ray Imaging is one of the most widely used radiography techniques allowing the doctor to get a look inside the patient's body without performing any incision which helps them to diagnose, monitor and treat many medical conditions. As the name suggests, it uses X-rays which is a type of electromagnetic radiation which is not harmful to human tissue in any way. These X-ray beams are able to pass though your body and depending on the material they pass through, these rays are absorbed in different amounts. For example, dense materials such as bones show up as white in X-ray images while fat and muscle appear as shades of gray. It is a very quick and painless procedure and usually requires little to no special preparation on the part of the patient unlike MRIs which may require fasting or the ingestion of special fluids by the patient. This X-Ray technology has also been employed in other types of diagnostic procedures such as fluoroscopy, CT scans and arteriograms.

3.3 - DICOM Files

DICOM stands for "Digital Imaging and Communications in Medicine" and it is the industry standard that was created by the National Electrical Manufacturers Association (NEMA) to aid the distribution and viewing of medical images such as X-Ray Images, CT Scans, Ultrasound, MRI etc. A

DICOM is both a file format and a communication protocol which means it can store medical information and patient's information all in one file ensuring that all the data stays together.

A single DICOM file contains a header (which stores information like the name of the patient, type of scan, dimensions of the image etc) and image data which can also be in three dimensions. This is better than the older Analyze format which stored this information in two different files (image data in a .img file and header data in a .hdr file). DICOM files also support compression. Since this file format is now the industry standard when working with medical images and is widely adopted by the hospitals, doctors and researchers can now easily exchange medical images with each other without any compatibility issues.

The DICOM files can be viewed using a DICOM image viewer. Most of the softwares that are actually used by hospitals for the analysis of medical images are closed source and are not sold to normal consumers. However, there are plenty of open-source DICOM image viewers which can be downloaded for free by people who want to view their medical images and analyze them at the comfort of their home. MicroDicom is one such open-source software that can be downloaded for free and that's why we have decided to use it and integrate it with our system to perform various tasks related to X-Ray images.Here is a flowchart to show the functioning of our system.

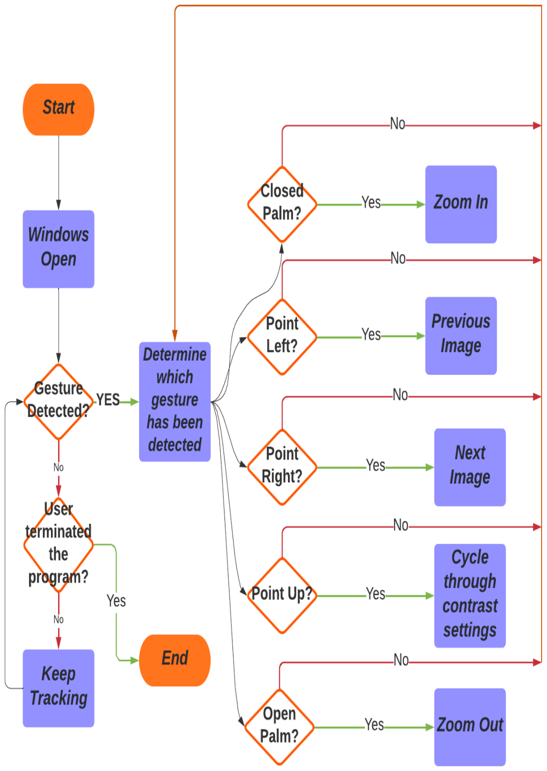

Figure 3 - Functioning of system

4. Result and Discussion

When our code is run, our trained CNN model is loaded and a message is printed in the command prompt to give the user an indication that the execution is in progress and the program is working correctly. After this, a couple of small windows pop up giving the user an idea as to what the webcam is capturing and how it is processing the frames to recognise the gestures. The name of the windows are as follows :-

- Trackbar
- Frame
- Processed
- Cam

The **Trackbar** window can be used by the user to adjust various settings using sliders so as to tweak the system to the lighting conditions around them for better and more accurate gesture recognition.

The **Frame** window shows the user what the webcam is capturing and gives the user an idea on how to position their hand for the best possible experience.

The **Processed** window gives the user an insight into how the system is processing the frames captured using the webcam. This is mostly used for debugging purposes and is not essential for the user experience.

The **Cam** window is the most important window out of the bunch. This is the main window that the user will be interacting with to input their gestures and perform tasks. It consists of a small green rectangular box where the user is supposed to perform the gesture they want to. Only the hand gestures inside the rectangular frame are tracked and any movement outside the rectangular box is ignored. This ensures that the system does not accidentally register commands and perform tasks that the user does not intend to. This window also has a number indicator at the top left. In our system, each gesture is given a number as follows :-

S.No.	Hand Gesture	Number Assigned
1.	Closed Palm	0
2.	Point Left	2
3.	Point Right	3
4.	Point Up	4
5.	Open Palm	6

Table 1 : Table showing what number is each gesture assigned

Whenever the system detects a gesture inside the rectangular frame, it updates the number indicator accordingly to tell the user as to what gesture is the system recognising. It also gives the user an idea if the system is detecting the gestures accurately or not. If not, the user can adjust the sliders in the Trackbar window for better gesture detection and accuracy. The number indicator displays 5 if no gesture is detected within the rectangular frame. This is the NONE state of our program. In this state, no actions are performed and the system stays as is.

Each gesture is mapped to a particular action that allows the user to control the X-Ray images within the MicroDicom software. Below is a table that shows which action is performed by which gesture :-

Hand Gesture	**Action Performed**
✊	Zoom In
☜	Previous Image
☞	Next Image
☝	Cycle through contrast settings
✋	Zoom Out

Table 2 : Table showing which gesture performs which action

As soon as a gesture from the above table is detected by the system, it performs the action corresponding to it inside the MicroDicom software. In order to make the actions more controllable and less eccentric, an artificial delay has been added between every consequent action so that the user gets the time to precisely control what they want to do with the X-Ray images.

In order to determine how well our system is performing, we did testing and documented the results. In our tests, we performed each gesture 100 times under ideal lighting conditions with the Trackbar settings set to their default values and we noted the number of times each gesture was recognised successfully to perform the assigned action. The success rate and failure rate for each gesture was then calculated. The results have been documented below with the formulas :-

$$\textit{Success Rate \%} = \frac{\textit{Action performed successfully} \times 100}{\textit{Total No. of times the gesture was performed}}$$

Failure Rate % = 100 - Success Rate %

Hand Gesture	**Action Performed**	**Success Rate**
✊	Zoom In	94 %
☜	Previous Image	95 %
☞	Next Image	91 %
☝	Cycle through contrast settings	90 %
✋	Zoom Out	96 %

Table 3 : Success Rate of each gesture

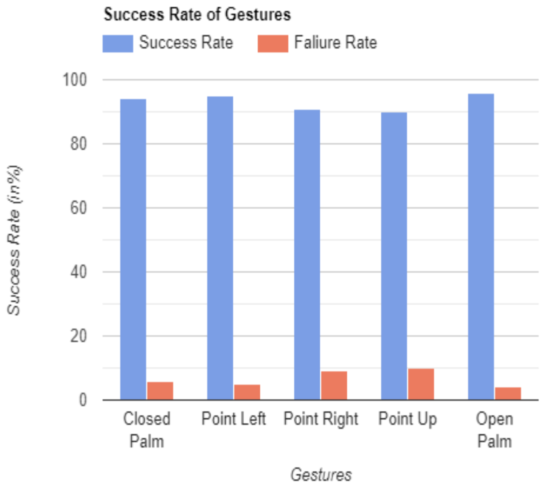

Figure 4 - Success rate of gestures

5. Conclusion and Future Work

The authors have built a gesture based system for controlling X-Ray images. We hope that our project helps a lot of doctors to perform their tasks more effectively while keeping them free from the risk of contamination especially in the light of the current Covid situation.

In the future, more hand gestures can be added and applied as input to the computer. These hand gestures can then be converted into computer commands to perform tasks in real time. Since our code will be flexible, it can be adapted so that it can be used in any application and it is not just limited to X-Ray image viewing. The gestures can be mapped to various keyboard commands and these commands can then be used to implement various operations within different apps.

6. References

[1] Hamida Amer, Ahmed Atia, Khaled Tawil : *Bacterial Contamination of Computer Keyboards and Mice in University and Hospital Settings*

[2] Maureen Schultz, Janet Gill, Sabiha Zubairi, Ruth Huber, Fred Gordin : *Bacterial Contamination of Computer Keyboards in a Teaching Hospital* **24** , 4 (2003)

[3] Dumitru Onceanu, A. James Stewart : *Direct Surgeon Control of the Computer in the Operating Room*

[4] Jiaqing Liu, Ryoma Fuji, Tomoko Tateyama, Yutaro Iwamoto, Yenwei Chen: *Kinect-Based Gesture Recognition for Touchless Visualization of Medical Images* **9** , 2 (2017)

[5] Ben Joonyeon Park, Taekjin Jang, Jong Woo Choi, Namkung Kim : *Gesture Controlled Interface for Contactless Control of Various Computer Programs with a Hooking Based Keyboard and Mouse-Mapping Technique in the Operating Room*

[6] Munir Oudah, Ali Al-Naji, Javaan Chahl : *Hand Gesture Recognition based on Computer Vision : A Review of Techniques* (2020)

[7] Soeb Hussain, Rupal Saxena, Xie Han, Jameel Ahmed Khan : *Hand gesture recognition using deep learning* (2017)

[8] G.R.S. Murthy & R.S. Jadon : *A Review of Vision Based Hand Gestures Recognition* **2** , 2 (2009)

[9] W.T. Freeman, K. Tanaka, J. Ohta, K. Kyuma : *Computer Vision for Computer Games , IEEE 2nd Intl. Conf. on Automatic Face and Gesture Recognition, Killington, VT, October,* (1996)

[10] Aditya Ramamoorthy A, Namrata Vaswani A, Santanu Chaudhury A, Subhashis Banerjee B : *Recognition of Dynamic Hand Gestures ,* (2002)

[11] Sayem Mohammad Siam, Jahidul Adnan Sakel, and Md. Hasanul Kabir : *Human Computer Interaction Using Marker Based Hand Gesture Recognition ,* (2016)

[12] Aashni Hariaa, Archanasri Subramaniana, Nivedhitha Asokkumara, Shristi Poddara, Jyothi S Nayaka : *Hand Gesture Recognition for Human Computer Interaction , 7th International Conference on Advances in Computing & Communications, ICACC-2017, 22-24 August 2017, Cochin, India* (2017)

[13] Dr R Suguna, Neethu P S : *Hand Gesture Recognition using Shape Features* **117** , 8 (2017)