

**Augmented Reality based Hand Gesture Recognition Towards 3D Geometry**

**Project Thesis**

Submitted By

|  |  |
| --- | --- |
| **ID** | **Name** |
|  |  |
|  |  |
|  |  |
|  |  |

**Department of Computer Science**

**Faculty of Science & IT**

**American International University Bangladesh**

## February, 2019

## Table of Contents

**Chapter 1: Introduction 6-9**

1.1 Introduction 6

1.2 Research Background 6-7

1.3 Problem Statement 7

1.4 Scope of the Research 7

1.5 Objectives 8

1.6 Significance of the Research 8

1.7 Research Outlines 8-9

1.8 Conclusion 9

**Chapter 2: Literature Review 10-17**

2.1 Introduction 10

2.2 Core Background Study 10-11

2.3 Review Based on Methods 11-13

2.4 Review Based on Frameworks 13-15

2.5 Review Based on Experimental Evaluation 15-17

2.6 Observation and Discussion 17

2.7 Conclusion 17

**Chapter 3: Research Methodology 18-22**

3.1 Introduction 18

3.2 Proposed Research Methodology 18-22

3.2.1 Recognition Using EmguCV 18-21

3.2.2 Recognition Using OpenCV 21-22

3.3 Conclusion 22

**Chapter 4: Research Methodology 23-28**

4.1 Introduction 23

4.2 Experimental Results 23-28

4.2.1 With Tracking 24-25

4.2.2 Without Tracking 25-26

4.3.3 Comparison of experimental results with previous research results 26-28

4.3 Conclusion 28

**Chapter 5: Conclusion and Future Work 29-30**

5.1 Introduction 29

5.2 Contribution of the research 29

5.3 Future Work 30

5.4 Conclusion 30

**References** **31-34**

### List of Tables

**Table 1-A** Selected methods with reference number 7

**Table 1-B** Research Outlines 8-9

**Table 4-A** Performance comparison (EmguCV) 25

**Table 4-B** Performance comparison (OpenCV with NumPy) 26

**Table 4-C** Performance comparison between modified method and previous method **27**

### List of Figures

**Fig. 2-1** The framework of review procedure 10

**Fig. 3-1** Extrusion Technique Architecture19

**Fig. 3-2** Modified Extrusion Technique Architecture 19

**Fig. 3-3** (a)Binary Normalization (b) Morphological extraction 21

(c)Background Subtraction

**Fig. 3-4** 0 blue Dot in 1, one blue Dot in 2, two blue Dot in 3, three blue Dot in 4, 22

four blue Dot in 5. (blue dot indicates the defect between fingers)

**Fig. 4-1** Enlisted Hand Posture Example 24

**Fig. 4-2**  Accuracy among the modified techniques and previous technique 27

**Fig. 4-3** Error among the modified techniques and previous technique 28

**Chapter 1: Introduction**

## 1.1 Introduction

From the very beginning of educational ascension mathematics has taken a special division. Geometry is one of the most affluent area of this. But from the very early stage, geometry learning has become difficult with the traditional tools and methods. Learners from preliminary stage get phobic about geometry. Many aspects like 2D and 3D plan, trajectory remain obscure to them and this reflects in their future learning process and potential. They lack of knowledge of those particular area and as the fundamental remain weak, the next phase started to be harder to them and in future the find whole mathematics as with complexity and fear.

Augmented reality has becoming more and more popular to the generation, what actually make the researcher to think that augmented reality can be a better medium of learning. What augmented reality does is mix the real world with virtually created object and give a better way to that object so that it seems the user is engaging with real time object. Now a days many games and application based on augmented reality ids found and children are interacting with them well. A tool like device is needed to interact with the augmented object. So now the focus is to use the technology based on Hand Gesture as the hand works like the best interactive tool with real time object and gesture is natural. But the main impediment to this field is different shape, color, posture of hand. So many researchers come with many proposed ideas to recognize the hand gesture with maximum efficiency. Moreover, the research in educational discipline also trying to ensure that both the student and teacher can interact, so that the learning can be more fruitful. Distance is also a great thing to deal cause student from different distance should participate in interacting the augmented object.

**1.2 Research Background**

Till now there are very few works hardly found worked fully with geometry and augmented reality. So, the field is yet unirrigated. Augmented reality can be chosen as a best medium of learning. A recent survey showed that the pupils who are less interactive to the class, they are found in the successor group of class when the medium turned into augmented reality instead of typical teaching method. Additionally, the good performer is assured with a high-performance teaching system which really gives a fruitful outcome. The distance between the students is narrowed down. In most of the education system followed a simple rule more or less. In any kind of teaching system, they first give the theory first than come the implementation or practical. For that the outcome remain less fruitful. In terms of augmented reality both the theory and practical work parallelly. So, the teaching become more efficient.

Before starting our research, we go through some papers to catch up with the recent research in augmented reality in education special mathematics, more specifically geometry. We selected some method to overview and then we go through the paper with some prospective of method, framework and accuracy acquired. Besides it should be mentioned that the augmented system must be low cost, good accuracy grabber, interactive and easy to use for the user especially preliminary school going children.

**Table 1-A:** Selected methods with reference number

|  |  |
| --- | --- |
| The “extrusion” technique [16] | Neural networks [5] |
| Construct3D (robust dynamic 3D geometry software in Augmented Reality) [35] | Baseline CNN Architecture: CNN and Adapted CNN architecture [22] |
| Fingertips tracking [15] | An auto encoder model [17] |
| Trajectory based method [7] | HOG-LBP Featured model. [21] |

## 1.3 Problem Statement

Developing a system which will recognize the hand gesture and generate 3D geometry using augmented reality. The system will be interactive for the students and the teachers in both ends. It will give the user to certain level of flexibility to control the augmented object.

There is a technique called extrusion technique [5]. By modifying the extrusion technique, we want to increase the accuracy and reduce the complexity of the system. Parallelly recognition rate increment in complex background is also desirable.

## 1.4 Scope of the Research

During the research we are planning to follow some steps to produce a good result or conduct a constructive and comprehensive research output. Existing research work review will be our first step. Extracting data from those research paper with some view of points we will trey to make a new method or will develop an existing method to its next level.

We will apply our own dataset or existing dataset on the methods to make a comparison among the methods to choose the suitable methods to work on. We will validate our hypothesis with modeling of a concrete prototype. Afterwards we will make comparison among the existing method and newly developed method and analyses the system and its findings.

## 1.5 Objectives

Augmented reality-based application is not an unknown matter now a days. Certain application can be found that will give us the taste of augmented reality. Our aim is to make an application based on Augmented Reality for 3d geometry by gesture recognition using hand. But it is unaware to us that specifically any application or system that directly helps tom learn geometry. Several apps found that they provide a limited benefit like fundamental view. But they are not interactive. Moreover, hand gesture recognition using Augmented Reality hardly found. Many procedures have been followed up by traditional method like deep learning or modified neural network but they are not implemented using AR for interactive 3D geometry building. As the time goes it become more and more essential to invent an interactive system based on modern technology and it can be said that Augmented reality is the best option.

We aimed to make an application that can recognize the hand gesture primarily. Our next objective is to produce 3D geometry using augmented reality. The system should be interactive in both ends. Additionally, the object should be controlled by the user like zooming, pinching, rotating etc. Along with the feature implementation it should be keep in mind that the system should not exceed a certain amount of cost.

## 1.6 Significant of Research

Existing methods found is the region of gesture recognition are mostly complicated and they partly give the user a full controlled set up. Additionally, the cost of implementation of the system is a considerable thing. Our aim is to build a cost-effective system by reduce the complexity. Using external device can help to detect the hand but if we can use a pre-installed webcam to surveillance the system that will be more preferable. Moreover, it will give the user a interactive system. So, the learning process of geometry will be more fruitful and effective. The education process will accelerate and the foundation of basic mathematical learning will be stronger.

## 1.7 Research Outlines

**Table: 1-B:** Research Outlines

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Timeline | B.Sc. Thesis | | | M.Sc. Thesis |
| Task | Week 1-12 | Week  13-24 | |
| Review Existing Methods |  | |  |  |
| Choose an existing method and work on that (Gesture Recognition Only) |  | |  |  |
| Validate the method with dataset  (Gesture Recognition Only) |  | |  |  |
| 3D geometry Build using Augmented Reality |  | |  |  |

## 1.8 Conclusion

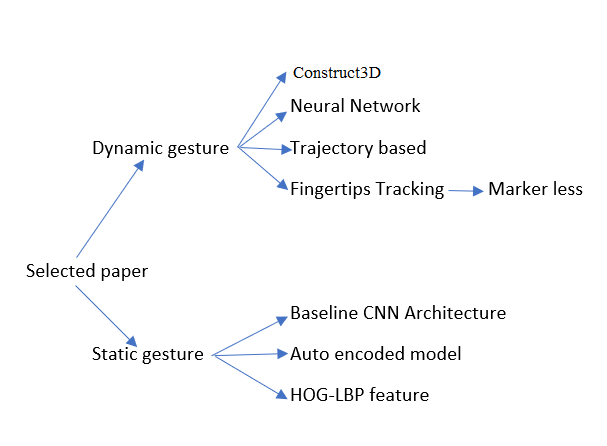
The field of augmented reality is still in its infant phase. Lot of works can be found that only process the hand gesture like Convolutional Neural Network, Hidden Markov Model, Deep Learning and so on but their research stays in the recognition only. Our aim is towards the augment reality. So, we are optimistic to build a system with augmented reality to take the learning process one step ahead. If one part of education system can be learned through augmented reality than other parts of the experiments can also be converted to augmented reality gradually and they can be controlled with gesture recognition.

**Chapter 2: Literature Review**

## 2.1 Introduction

Augmented reality is a new technological milestone. This technique is new to the researcher. So, several researcher working relentlessly to produce a good output. They worked with several methods like Convolutional Neural Network, Hidden Markov Model, Deep Learning, Extrusion technique. Several researchers proposed several frameworks. Some proposed a new method, some modified the existing method. The validated their own method by existing dataset or self-made detest. Some deals with static dataset where a very few people choose dynamic dataset. As a consequence, the outputs were different, the accuracy were different. Very few methods deal with augmented reality. So, considering all these factors a constructive literature review is done to choose a suitable method to work on. They were cross examined with methods, framework, dataset and experimental results. As a output of the review a bunch of method was selected according to the criteria and among them finally a method was chosen to work on.

## 2.2 Core Background Study



**Fig. 2-1:** The Framework of Review Procedure

The first augmented reality-based application was a head mounted display invented by Ivan Sutherland paved the way of AR. It was named The Sword of Damocles in 1968. In 1972, another invention named Video place developed by Myron kthe first augmented reality-based application was a head mounted display invented by Ivan Sutherland paved the way of AR. It was named The Sword of Damocles in 1968. In 1972, another invention named Video place developed by Myron Kugar was a milestone. It gave the user an interactive environment. In 1998 AR was used for navigation by NASA in their X-38 spacecraft. A game named AR Quake was the first AR based game. Uses of AR in smartphone started in early 20s. Many worked has done to establish augmented reality in various discipline of education like medical, engineering with limitations.

Now a days the main goal is to increase and use the spatial ability. Spatial ability presents an important aspect of human intelligence. There are five things that can be the fundamental blocks of spatial ability are perception, visualization, rotation, mental rotation and orientation. In recent studies it has been verified that spatial ability can be improved by geometry education. 3D modeling with 6DOF (Six Degree of freedom) has been taken as a massive sector of research work. In recent years there are immersive works related to pure education using Augmented Reality. It is mentionable that CyberMath [1] is an avatar-based application that helps to learn mathematics but remotely not interactively and desktop dependent. Other desktop-based application like Geometer’s Sketchpad [1], Cindarella [1], Euklid [1] and Cabri Geometry [1]. They intendent with dynamic support but only for two-dimensional geometry.

In 2013, Volkswagen utilizes increased reality as their vehicle manuals. The utilization of the MARTA application through the iPad can enable clients to see the inner functions of the vehicle so the administration mechanics comprehend what they are managing. The MARTA application can likewise indicate progressive directions to assist mechanics with the tasks they are chipping away at. It can even help with supplanting parts. It can even be as explicit as which bearing the parts ought to confront. The MARTA application can be utilized for more corrective tasks too, including perceiving how extraordinary shading paint employments can look on your vehicle. In 2014, the Google Glass is uncovered and is made accessible for shoppers. The Google Glass wasn't as fruitful as designers trusted it would be, yet it showed the capability of what wearable expanded reality could be. The second emphasis as of now is by all accounts all the more encouraging and more helpful. Rather than utilizing the Glass to look through internet-based life and different applications, assembly line laborers are utilizing the innovation to help with regular work. It helps walk the laborers through their everyday undertakings and be more beneficial and productive. In 2016, Microsoft presents the following emphasis of wearable expanded reality. The HoloLens is by all accounts everything that the Google glass needed to be, however surely not as cautious and wearable in regular daily existence and is without a doubt costlier. The innovation progression between the two is obvious, however the value scope of $3000 and $5000 are out of a great many people's financial plans.

## 2.3 Review Based on Methods

Firstly, we were introduced with the method “*The Extrusion Technique*” which creates 3D shapes from 2D shapes but complex for some shapes (e.g. circle) [16]. In another research a different method came to our attention which is “*An auto-encoder model based on Semi-supervised Learning*”. As the name suggest it is based on semi-supervised learning but only for static gestures. Dynamic gestures are often a contiguous sequence for it which makes it inadequate for real-time use [17]. In the following paper a comparative methodology is utilized which is "K-Means grouping" which produces bunch an incentive at the nearest remove and gets another method for each group, however dependably relies upon past group esteems. For this, additional tool named *Support Vector Machine (SVM)* is used [18]. “Masking and Gradient Approach” was introduced in next research. It depended on every single pixel point of an image. That means regarding data storage it has an advantage but regarding high resolution images it will not work well [19].

A hand acknowledgment strategy dependent on HOG-LBP intertwined highlights and the spiral base capacity bolster vector machine as the bit work for ordering hand signals on MATLAB utilizing LIBSVM tools. It focuses on the static gesture recognition technology and has complexity in dynamic gesture [21]. Adapted Convolutional Neural Network for ordering static hand motion picture informational collections shifting in lighting, clamor, scale, rotation and interpretation [22]. The proposed calculation utilizes an image-based analysis and put together and the examined with respect to the different human signals a man can involvement in regular daily existence. The focus is on detecting and recognizing manual movements in a natural environment [23]. A method was introduced to have different hand gesture templates at the beginning of the system and to evaluate the best match. The system is suitable in real time with an embedded device which has lower processing speed than a personal computer. [24].

Bidirectional rank pooling method converting the depth sequences into images that improves the recognition accuracy largely [25]. Dynamic hand motion demonstrating to delineate subset of signals to a significant framework order. The proposed iterative plan is computationally proficient in examination with the generally utilized channel-based quadrature techniques, in spite of the fact that it is a long way from real - time individual can involvement in regular daily existence, but it is sensitive to the initial selection of real image sequences [26]. A method for recognizing gestures with RGB and depth data on a 3D coevolutionary network. The recognition of the gesture can be distracted by the color of the skin and attire [27]. A strategy called Shadow Catcher where shadow-based light recuperation should be possible with a generally little blocking article that is less meddling and can be expelled all the more effortlessly from the last scene later. These strategies work predominantly with hard shadows, since edge recognition techniques break with gentler shadows intrinsically [28].

Freeform modeling approach which is a Windows®-application written in C++ and can also easily be ported to Linux® and Mac OS® X [29]. A collaborative experimental hybrid user interface that combines AR, conventional 2D GUIs and all - round computer elements. Using an environment 3D model, indoor users create virtual objects and real objects are highlighted for outdoor users to visualize and preserve track of the activities of outdoor users, while users of outdoor will highlight intriguing objects and events for indoor users to observe. [31].

Azad el at. [2] Used depth motions technology for hand gesture recognition which deals with inter-class and intra-class and works in dynamic state. EMG signals were taken as input and SVM and CNN is used as a classifier for higher accuracy by CHEN et al. [3]. A freehand technique was proposed for manipulating distanced object by Jung et at. [4]. Neural network is adopted by Munasinghe [5] for gesture recognition and in his method, there are a secondary option if the Neural network failed to classify the hand gesture. Parallelly Alam et al. [6] also deal with neural network but his procedure found to be more complex. Liu el at. [7] Offered a trajectory-based method that has proven to be faster than the vision-based method. Krupka et al. [8], Rani et al. [9] both taken vision-based method for their recognition where first one is marker less and second one is capable of learning without any pre-requisite algorithm. Chiang et al. [10] consider depth information processing for hand gesture recognition and the greatness of hiss process is that it is void of palm width calculation. Kitimat et al. [11] and Liu [12] worked with combination of inertial sensor and depth sensor and only inertial sensor respectively. These processes don’t cost much for hand gesture recognition. Hidden markov model is adopted by Liu et al. [13] which is also a low-cost process, but it should be mentioned that it contains some unobserved factor and success depending on existence of unobserved factor. Gudavalli et al. [14] takes an initiative to cascade multiple method for highest recognition accuracy and it’s totally dynamic. A real time marker less method is approached by Oka et al. [15] and their method is depending on recognition of the finger tips.

A three - dimensional geometry building tool that build geometry dynamically named Construct3D that is mature enough to be used in educational practice. [33]. The main benefits of using Construct3D in secondary school geometry education is that students actually visualize three - dimensional objects with traditional methods [34]. It is easy to use, takes little time to learn, encourages students to investigate geometry and can be used consistently. The optical tracking system is expensive [35]. Construct3D brings together four areas of research: geometry, pedagogy, psychology and increased reality. The advancement of considerable instructive substance will require a comprehensive assessment of the real value of an education tool. [36].

## 2.4 Review Based on Frameworks

A webcam perceives the AR markers while the AR Toolkit libraries overlay and show the 3D virtual questions on the physical AR markers in the AR condition precisely [16]. Convolution and pooling are utilized to extricate highlights from the first pictures of the signal. The SoftMax classifier is then used to actualize the characterization and streamline the parameters of the whole system to enhance the order exactness [17]. A personal computer (PC) is used to track the skeleton and to process the recognition of gestures. The sensor uses Microsoft Kinect v2 as an application development program for Microsoft Visual Studio 2015. Windows Sensor Kinect v2 can be developed in different installations. The use of Kinect v2 camera and proximity sensor functions is used to obtain a XYZ coordinate axis and calculate preprocessing using statistical data. [18]. The calculation of the palm width is necessary to obtain the finger width. The width of the finger is between one third and one fourth times the width of the palm depending on the finger extension. A disintegration channel is then built up that is a piece of the finger circle measurement. This channel is utilized for disintegration of the divided twofold picture. Enlargement of this picture adds to the cover (palm here) [19].

Construct3D initializes a 3D window with the maximum size to cover the virtual space at the very starting of the system. The UI is instated and the menu framework is mapped to a handheld board called the PIP. Haptic criticism from physical props manages the client while interfacing with the PIP, while the illustrations overlaid enable the props to be utilized as multi - utilitarian instruments [34]. The standard immersive configuration used for Construct3D supports two collaborating users with stereoscopic head - mounted displays (HMDs) that provide a shared virtual area. Users interact with the system through pen and pad accessories [35].

Azad el at. [2], they use a mixed strategy of (Multilevel Temporal Sampling +Weighted Deep Motion Map). The input data is augmented using MTS and HOG is used to describe the local object. WDMM is used to distinguish the motion. The framework has only disadvantage of having too much computational step. Using neuromuscular activity property CHEN et al. [3] uses EMG signals as input in their framework and the classification is done alongside with support vector machine. Due to weight sharing technology it is a parameter reduced framework. It works well with nonlinear high dimensional image but only steady EMG signals are considered. Jung et at. [4], which technique they apply actually a modification of GoGo system that is not mapping dependent. It through exception when trying to rotate and fingers are too close. Munasinghe [5] organized his framework by following steps are background segmentation, removal of noise, applying binary thresholding and classification where Alam et al. [6] followed multilayer perceptron classifier and centroid classifier. But it can recognize only 10 static hand gesture.

Framework is subdivided into two phase named training phase and recognition phase by Liu el at. [7]. As the validation done by ten cross validation, the accuracy rate is higher and the computational step is low in this framework. In Krupka et al. [8] framework is so simple image capture, hand detection, hand gesture detection in three different phases. It has lowest false positive value and accuracy but it deals with only static hand gesture. Rani et al. [9] framework is quite complex started with EMG signals and they used neural network too. Computational complexity is reduced here and it is implemented in real time.

Chiang et al. [10] uses palm contouring and from that extracted feature are used for gesture recognition. Kitimat et al. [11] and Liu [12] both deals with vision based and their framework consists of feature level fusion, decision level fusion, pooling, classification and gesture recognition. Three-fold framework is designed by Liu et al. [13] with language proposition and basically designing an algorithmic pipeline. Gudavalli et al. [14] has started their framework from video inputting than cascading analysis of motion, shape location and classification is followed by gesture recognition.

The spotting system is assessed on two datasets with four folds from the acknowledgment dataset each. Each spotting informational collection crease is made by the relating acknowledgment dataset overlap. For each client, each crease has 60 recordings. Of every client's 60 recordings, 20 recordings are made by arbitrarily choosing 3 recordings from the acknowledgment dataset, 20 recordings are made by haphazardly choosing 5 recordings and 20 recordings by choosing 7 recordings. This outcomes in a sum of 660 recordings and 3300 signs for every crease. Dataset-2 is likewise made, but random frames are inserted between gestures. [38]. Fan Zhang [39] follow the framework in Fig 1. Dinesh Kumar Vishwakarma [40] divided the process into four parts: Depth Detection, Contour Extraction, Convex Hull and Defects and Tracking and Finger Detection. Tanatcha Chaikhumpha [20] use 3 step frameworks. Step 1: Hand Segmentation & Preprocessing, Step 2: Hand Tracking & Feature Extraction and Step 3: Gesture Classification [20]

## 2.5 Review Based on Experimental Evaluation

Unnamed dataset was used in extrusion technique and the parameter used is the relationship between the geometric forms. Fingertips method used their own made dataset and it used Linux OS, Pentium iii 500 MHz, Hitachi iP5000 image processing board, Nikon laired-s270. In trajectory base method self-made dataset were used and the parameter used to perform the experiment were Acer 5750G notebook, Intel corei5-2450M, 2.5 GHZ, 4GB ram, Win & DEV C++ and C, OpenCV 2.1. Neural Network based method also used their own dataset and their parameter were Corei7(4720),2.6 GHz, 8GB ram. Baseline Convolutional Neural Network Architecture select Hand gesture dataset of LSP as their dataset and the parameter was total of 10 gestures that match the pinch, scale, buckle, hold, grab, rotate, crawl, OK, cut, shoot. Auto encoded model chooses confusion matrix of the classification as their dataset and various scale, pinch, rotation was taken as parameter. NSU hand posture dataset was taken by combination of Histogram of Oriented Gradients (image processing) and Local Binary Pattern method and the parameter was the accuracy of recognition of manual gestures is tested on subsets.

The AR application took 53.8 seconds to finish the assignment and GeoGebra took 71.95 seconds. Conversely, Cabri3D took 47.67 seconds since this is an extremely straightforward application [16]. The disarray network of arrangement demonstrates that the blunder rate was fundamentally due to a 15% of the signal clasp mistake partitioned into the wrong motion cut, while 24% of motion shoot mistake isolated into the wrong motion squeeze, from the first database pictures can be seen, the principle reason is that the high closeness edge highlight of these pictures. It drove along these lines to the ultimate result of disarray [17]. Execution process utilizing K-Means takes longer than utilizing the SVM strategy, however the K-Means precision rate is lower than the SVM technique. [18]. 50 video samples for training and testing by 4 people each test 8 moves and 1 moves 50 times. The average rate of recognition is 96.25%. [20]. Experiments with a set of five people with different skin tones led to the following results. [19]. Understudies needed to work in one of the instructional meetings: Students need to construct a progressive surface by turning a B-Spline bend (cubic, 5 - 6 control focuses) around the pivot on the hub. [18]. There have been significant positive correlations between DAT: SR and PSVT: R on the one hand and non - verbal reasoning on the other [19].

Azad el at. [2] Make three types of their input video in long, medium and short. Feature extracted using VLAD. Classification were done through combinedly SLNF and ELM. The experiment was applied to four data set and they showed 98.05, 97.31, 95.24, and 68.66% respectively. Taking EMG signals as input CHEN et al. [3] extracts feature using convolutional neural network and then classify them using SVM classifier. There are totaling 11 layer and the experiment was done on Nina pro database and it results in 65.4 and 68.2 % on CNN and SVM respectively. Leap motion is used as tracking technology by Jung et at. [4] And virtually hand is managed in two mode. It also provides a system to reset the. Self-made dataset left GAS 4.7 by accuracy. Munasinghe [5] applied his experiment in self-made database with Gaussian mixture-based subtraction, median blur and thresholding. He tested in low light and good light results in 71.3 and 85% respectively. With a convolution layer size containing 300 layer and dropout 0.4% Alam et al. [6] applied their procedure Khushboo el data set and the accuracy rate are at highest peak 98.88%. Liu el at. [7] used 2 cross and 10 cross validation to train and applied it on self-made database and he got 92.3 and 94.7% accuracy. Using IP camera hand contour was extracted using convex hull detection and applied the experiment on a self-made database and the outcome in 10 different gesture. Average accuracy was 90% in Rani et al. [8] self-made database. Hand posture pipeline and pose estimation was applied with CTE value 290. Mentionable steps in Chiang et al. [9] framework Connected component learning, contour smoothing, convex hull and blending hand posture. The experiment was performed on a self-made database. For index finger the result 92.12% and for two fingers 98.87%. Kitimat et al. [10] used two types of method for his classification. First one is multilayer perceptron and second one is k-means classifier. Average accuracy found 93. Two type of wrapping DWT and elastic matching was used as matching technique in Liu [11] experiment. After particle filtering hidden markov model is used for classification on ten gesture of Microsoft MSR dataset and the acquired accuracy is 93%.

Liu et al. [12] was aimed to find out the best classifier. So, he fed into the input in HMM and then classify the input using gyroscope, accelerometer and Kinect. The experiment is tested on $1Unistroke recognition application and the accuracy rate is 91%. Extracting gesture from multiple sequence of gesture in Gudavalli et al. [13] experiment. Than extraction of motion component and location component is performed using PCA and PWDTW. Additionally, shape component is extracted using STIP. The experiment is tested on CHEALEARN gesture dataset and from the dataset, every time the cascading method take least time than other traditional method. Detecting multiple figure is an important feature of Oka et al. [14] experiment. After measuring fingertips trajectories gesture is recognized with thumb detection and symbolic gesture recognition. The experiment was performed on a self-made dataset and there are two types of accuracy. For sing le finger accuracy rate is higher counting 99.2% than double finger accuracy rate 97.5% though their difference is not that much noticeable.

Optical flow is extracted for each two consecutive frames and a 8-bin histogram is created based on the flow direction. [38] [19]. Ten pictures of ten people were taken. The segments of the skin used to calibrate our Gaussian skin model were taken randomly. Accuracy by masking 96 percent and 98 percent by gradient approach [39]. The experiments are carried out on the NUS data set-II hand posture. Chaikhumpha Tanatcha [20] proposed system offers good results in recognizing gestures in real time from color image sequences through the single-hand movement trajectory and multiple HMM usage. The HOG highlight has turned out to be outstanding amongst other highlights for securing data on the edge and neighborhood shape and has been effectively connected in fields, for example, target discovery and recognizable proof. The LBP include is an extremely successful surface component descriptor that is strong for changing the dim scale and changing the picture pivot [39].

## 2.6 Observation and Discussion

The findings showed that there are some tools and applications for education purpose using augmented reality. But they are not fully dynamic that means real time hand gesture recognition is hardly found. Most of the researcher deal with static hand gesture recognition. Very few works with partial dynamic gesture recognition. Marker is also an important fact. Most of the method used for hand gesture recognition, marker is used. But marker is kind of dependency injected and need to train up the device for recognition and marker-based method can produce a little bit of augmented object.

There is some factor like lighting condition. Our findings find only one to deal with the light condition and the result was excellent in low light condition too. In addition, in case of dynamic hand gesture recognition the data set become larger and larger. Than it needs a huge database. Complexity of the method and computational time both increase in that case. As we concern about education purpose, so there must be a way of interaction between the teacher and the learner on augmented object. But most of the published work is not interactive. So, what we need is really challenging in this stage of AR research. We need a fully dynamic, collaborative and interactive hand gesture recognition system for the education environment within low complexity and independent.

## 2.7 Conclusion

The review concludes that among the researcher gesture recognition is a common thing. Most of them just focus on the recognition of hand gesture. They applied their own method and experimented to recognize the hand gesture with minimalist complexity. Some of them use combined method to get better accuracy. “Extrusion technique” is found directly related to augmented reality so far. But they used an external device to recognize the hand. As we are thinking about real time gesture recognition so the data must be dynamic too. So static dataset got less priority in the review.

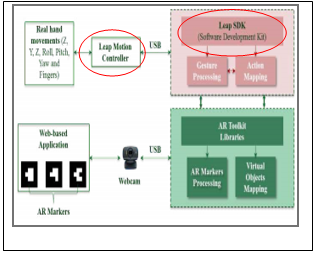
**Chapter 3: Research Methodology**

## 3.1 Introduction

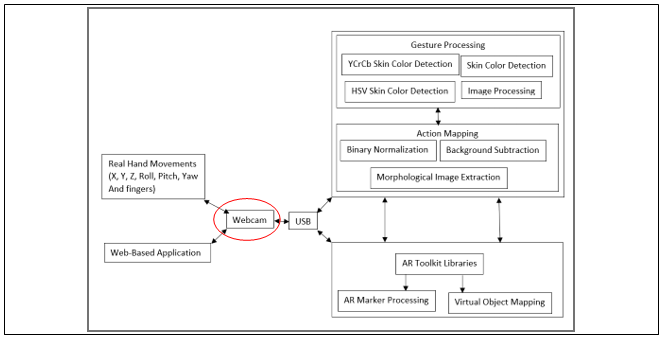
A variety of work is done on gesture recognition some researcher takes it to the further extenuation that they use the hand gesture to control an augmented object. Different researcher uses different kind of method like deep learning, neural network, Hidden Markov Model, Vision based method and so on. They use as less step as they can but to get a proper recognition some systems become lengthy like [12], [13] but they got higher accuracy. Alongside, some method got too much computational and complex like method of [6]. And most of the research work is done on steady and static gesture recognition. Very few methods like [8], works on dynamic hand gesture recognition and they work well. Most of the researcher uses self-made dataset. Among the various technique like auto encoded model, deep learning “extrusion technique” achieved highest priority in terms of dataset, framework, methods, complexity and how much it relates to the context. The main disadvantage of the technique is that it uses an external device called leap motion device. This device will recognize the hand and the recognition output will be action mapped. Using of this external device cost a little bit. So, we tried to modify the technique to reduce the cost and perform the recognition process using the webcam. We tried to increase the recognition accuracy as well. Instead of leap motion device our proposition was to use written code to recognize the hand gesture.

## 3.2 Proposed Research Methodology

The method essentially centered around two principle innovations Augmented Reality and hand gesture acknowledgment to develop a hands-on learning technique for the learner. With Augmented Reality, the learners can comprehend the fundamental ideas of 3D geometrical shapes, their connections and approaches to build the 3D shapes and the items in 3D space. Vitally, Augmented Reality can give a dynamic representation of 3D structures of geometrical shapes. This element causes the students to comprehend a complete foundation of 3D geometrical shapes and enhance the capacities of geometrical structures. In addition, the hand gesture-based connections outfit an instinctive and advantageous path for the understudies to specifically control and cooperate with geometrical shapes in 3D space.



**Fig. 3-1:** Extrusion Technique Architecture [16]



**Fig. 3-2:** Modified Extrusion Technique Architecture

With the encounters of interfacing with the 3D shapes utilizing their own hand motions, the understudies can enhance their very own consciousness of the connections of the 3D shapes and effectively recollect or hold the learning about the 3D shapes.

Architectural overview of the proposed system is shown in Fig. 2. The proposed system deals with dynamic hand gesture recognition, so this is a real time system. Various types of posture are enlisted for recognition, but firstly, only worked with the numbers of fingers to testify that whether the system can recognize hand or not. The system will visualize through a basic webcam to recognize the hand and hand gesture. Than it will bound the hand and extract the hand from other parts of body as the skin types are quite similar for hands and other parts of body. The student can visualize the geometry shape both in 2D and 3D with the assist of AR markers. Than the AR marker is obtained by the webcam. During the hand gesture recognition and the AR marker recognition, in both cases a webcam is used while the AR Toolkit libraries overlays and displays exactly the 3D virtual objects on those physical AR markers in the AR environment. Whenever the students or the teacher will try to interact the augmented shape, it will map the gesture and will produce pre-built geometrical shape in the augmented environment.

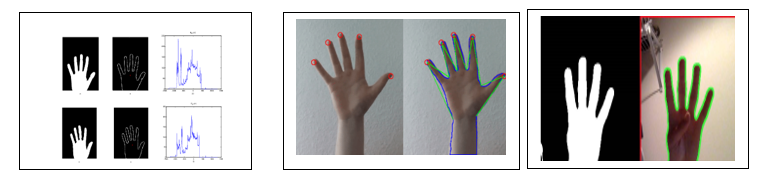
**3.2.1 Recognition Using EmguCV:**

1. **Step 1: Gesture Processing**

Recognition process starts with hand gesture recognition. User or the learner have to be available before the webcam that is installed on the personal laptop or computer. User can do move like raising finger starting from one to ten. They also can show their thumb finger according to the given posture at fig. 11. Hand gesture is recognized through the webcam. A series of hand-written code is used to recognize the hand gesture. The code is segmented is YCrCb skin detection, skin color detection and HSV skin color detection. YCrCb is a commonly used color space in digital video domain. Because the representation makes it easy to get rid of some redundant color information, it is used in image and video compression standards like JPEG, MPEG1, MPEG2 and MPEG4. In this format, luminance information is stored as a single component, and chrominance information is stored as two color-difference components (Cb and Cr). Cb represents the difference between the blue component and reference value. Cr represents the difference between the red component and a reference value. Skin Color detection deals with the recognition of skin-colored pixels and regions in a given image. Skin color is often used in human skin detection because it is invariant to orientation and size and is fast to process. The three main parameters for recognizing a skin pixel are RGB (Red, Green, Blue), HSV (Hue, Saturation, Value) and YCrCb (Luminance, Chrominance) color models. The HSV color space is more intuitive to how people experience color than the RGB color space. As hue (H) varies from 0 to 1.0, the corresponding colors vary from red, through yellow, green, cyan, blue, and magenta, back to red. As saturation(S) varies from 0 to 1.0, the corresponding colors (hues) vary from unsaturated (shades of gray) to fully saturated (no white component). As value (V), or brightness, varies from 0 to 1.0, the corresponding colors become increasingly brighter. The hue component in HSV is in the range 0° to 360° angle all lying around a hexagon. The main reason of the code segmentation is to recognize the hand fully. Because, there could be any kind of disturbance like same color things, other parts of the body. So, despite all these problems the problems the code can segment the hand fully and can recognize the gesture.

1. **Step 2: Action Mapping**

Action mapping starts with the ending of skin detection. The first step is generating a three-dimensional image space which is obtained by the two-dimensional binary image normalization. Than the background is subtracted using the median value applying a threshold in the color space. At the end a morphological image is extracted. Morphological image processing is a collection of non-linear operations related to the shape or morphology of features in an image. Morphological operations rely only on the relative ordering of pixel values, not on their numerical values, and therefore are especially suited to the processing of binary images. Morphological operations can also be applied to greyscale images such that their light transfer functions are unknown and therefore their absolute pixel values are of no or minor interest. Morphological techniques probe an image with a small shape or template called a structuring element. The structuring element is positioned at all possible locations in the image and it is compared with the corresponding neighborhood of pixels. Some operations test whether the element "fits" within the neighborhood, while others test whether it "hits" or intersects the neighborhood Than the action is mapped to a pre-defined model or dataset and the output is shown in the ImageBox Advanced gesture mapping methods can be used to manage a variety of gesture conflicts and gesture ambiguities, such as: pose ambiguity from tracking errors due to motion blur, hand self-occlusion or pose confusion from similarity and user error. When working with rich gestures (where high degrees of freedom are available) there is a greater chance of variability in the performance of gestures from one user to another. This is especially true of 3D motion gestures as there are fewer physical constraints on motion or pose, unlike surface touch gestures which limit motion to the plane of the 2D surface or hand-held gamepad controllers with mechanical buttons that limit motion. As a result, there is greater variation in user-performed actions (poses and motion) within bare-hand motion gestures.



1. (b) (c)

**Fig. 3-3:** (a)Binary Normalization (b) Morphological extraction (c)Background Subtraction

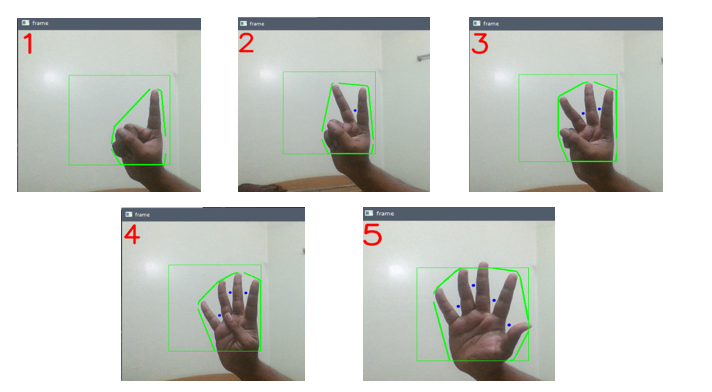
**3.2.2 Recognition Using OpenCV**

OpenCV (Open Source Computer Vision Library) is a library of programming functions mainly aimed at real-time computer vision. Originally developed by Intel, it was later supported by Willow Garage then Itseez. It has C++, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. We use python as programming language.

1. **Step 1: Gesture Processing**

At first in the program define region of interest, skin color in HSV. It then applies *GaussianBlur* OpenCV function for python to smooth the image. The function convolves the source image with the specified Gaussian kernel. Then it finds contour with maximum area and creates bounding rectangle around the contour to find convex hull and convexity defects. And then it shows the required images. Step 2: Detect Hand Action

1. **Step 2: Detect Hand Action**



**Fig. 3-4:** 0 blue Dot in 1, one blue Dot in 2, two blue Dot in 3, three blue Dot in 4, four blue Dot in 5. (blue dot indicates the defect between fingers)

When hand come in the target region, it finds the defects due to figures. Suppose there are zero defect on one finger, one defect between two fingers, two defects between three fingers, three defects between four fingers and four defects between five fingers, so after detect the defect count program show the correspondence output on frame using OpenCV *putText* function in python.

## 3.3 Conclusion

The development of an efficient hand gesture recognition system is an important aspect for easy interaction between man and machine. In our findings we came across many experimental steps to determine how efficient each of the methods and frameworks are for recognizing hand gesture. The main focus of these experiments was accuracy. Most of the experimental results found in the research were promising. Although, some have shown greater result than others, even the most effective one has some weaknesses. It seemed that these experimental results were more effective because they were taken in terms of static gestures. When it comes to recognition of dynamic gestures, the results show less accuracy then that of static gestures. But as for the other methods and frameworks despite of showing less efficiency in recognition of static gestures they show more efficiency in recognition of dynamic gestures then those mentioned before. Their accuracy rate is higher than the others’ accuracy rate which results shows efficiency in static gesture. And in the domain of augmented reality the results suggest the processing rate of the methods. They are based on specific task performance. They differ among the tools used for Augmented Reality.

**Chapter 4: Experimental Results**

## 4.1 Introduction

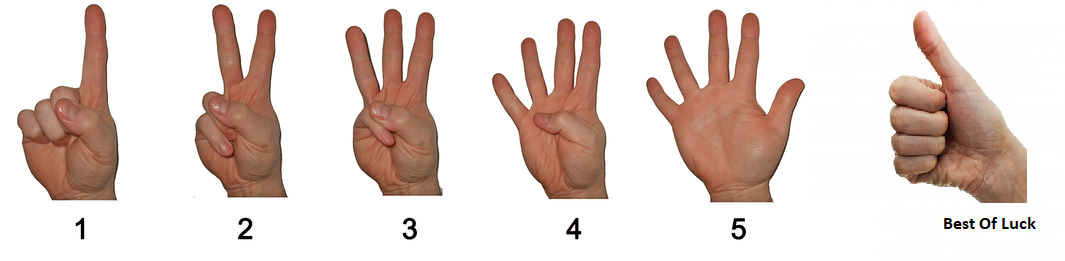
The experiment was occurred in home environment. We experimented on various background. Firstly, we experimented on simple plain background. Than we choose a background that’s match the skin color to examine that how perfect the process can segment the hand. Than we experimented on a very complex background consists of many things. Many things like window grill, wardrobe, flower vase, books and other domestic things as we experimented it at home environment. No external light is used to focus the hand. Typical room light was the light source. We experimented our proposed method in two way. First one is with tracking and second one is without tracking. With tracking can recognize the hand in any region and without tracking needs to be place the hand in a specific area. The IDE we used to build the program is visual studio enterprise 2017 edition. The program was running on a Lenovo laptop consisting of corei5 7th generation, 2.70 GHz processor and the operating system was Windows 10. A pre-installed webcam is used to recognize the hand. The program was developed using EmguCV. EmguCV is a cross-platform image-processing library [44]. It is closely related to OpenCV because EmguCV is a .NET wrapper to OpenCV. We can say EmguCV is OpenCV in .NET. In without tracking the program was written using openCV with NumPy along with OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library [45]. NumPy is the fundamental package for scientific computing with Python [46].

## 

## 4.2 Experimental Result

As we mentioned we were working with two types of program, we used the webcam to recognize the video. Four people were organized to perform the task that they will show their finger from one to one and some other posture. They sat before the camera and perform the task as instructed. As it is a real time application, so there is no still image dataset. We used our own self-made video dataset to recognize the hand gesture. Each were instructed to perform the task for five minutes. Five minutes were taken to provide them enough time to complete the task five times. They raise their finger from one to five in both hand and keep the hand steady for five seconds each. Five seconds were taken so that the webcam can recognize the hand gesture correctly. is They done the process five times? We repeated the process five times top find if there is any improvement in recognition or find specific reason of wrong recognition. We found texture of the background has a great impact on gesture recognition.

Our research is basically subdivided into three phases. They are recognition of hand gesture; recognition of AR marker and final step is to generate the geometrical shape with interaction process. As previously mentioned, we only worked with the numbers of fingers to testify that whether the system can recognize hand or not. So, we have conducted experiments on how accurately the program is recognizing enlisted posture. To further extend the results we have tried experiments with two types of program.



**Fig. 4-1:** Enlisted Hand Posture Example

**4.2.1 With tracking**

This program is able to track the hand from any position and extract it from other parts of body as the skin types are quite similar for hands and other parts of the body. It then puts AR markers around the hand. The approach shown is based on skin detection and convex hull and defects computation. So, for checking its accuracy we run this program in various backgrounds. The base of the program is implemented by Luca Del Tongo. In its base the program was able to take specified videos of Hand gestures as input. When we first collect the code, it only takes input from a pre-defined video dataset. We have modified it so that it would take input from webcam. There was a grabber function which takes input from video, we made it global to take input from any kind of desktop or laptop facilitated with webcam. Regarding Simple (Non-Complex) background the program is able to track and recognize the gesture. As the color pallet of the background is different from skin color pallet it shows no problem in tracking the hand posture. So, in terms of simple/non-complex background this program holds potential. Regarding Complex background, it was observed that the program had great difficulties in tracking the hand. Rather most of the times it would track the background as it may have similar color pallet that of skin. Hence it rarely gave the correct response according to the posture. Moreover, it overlaps the hand skin with the same color background and gives falls recognition.

Despite having no hand displayed on camera the program still responded in complex background because of matching color pallet with skin. According to the discussion above it is apparent that it has a high accuracy rate in recognizing and tracking hand gestures when the background color pallet didn’t match with the skin color pallet. But it was mostly unable to recognize the hand when the background color pallet matched with the skin color pallet.

**Table 4-A:** Performance comparison (EmguCV)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Background Type | Accuracy = Successful recognition/Total no of attempts | Error  = Unsuccessful recognition/ Total no of attempts | Tracking Rate = Recognized hand region /  Total hand region (A pre-defined threshold) | System |
| Simple (Non-Complex) | 90% | 10% | 65% | Core i5 CPU with 2.70 GHz processor |
| Complex | 20% | 80% | 10% |

In Table 4-A we illustrated what we have observed from the performance of the program. The program was run in Core i5 CPU with 2.70 GHz processor system. We operated the program in two different types of background. One was Simple background which did not have any large mix of color pallet and also did not match with skin color pallet. In this case we used white wall as a background as it is the simplest and the hand is in a great contrast with the wall to recognize. Another was Complex background which had large mixture of color pallets and/or matched with skin color pallet like doors or windows. And as mentioned before we raised our fingers from one to five in both hand and keep the hand steady for 5 seconds each. It was observed that 65% of the time the program was able to track the hand in Simple background. But it had a hard time tracking the hand in Complex background which was 10% of the time. For Simple background the program was able to recognize the gestures successfully 18 times but for Complex background it only successfully recognized 4 times.

**4.2.1 Without tracking**

This program is able to track the hand from a specific position where only the hand will be shown. The program will recognize the hand from a specified area hence it doesn’t need to track. So, for checking its accuracy we run this program in various backgrounds. The base of the program is implemented using Python 2 which is backdated. So, we have converted the program from Python 2 to Python 3 as it is more updated. This also opens more capacity for future improvement of the program.

Regarding Simple (Non-Complex) background the program is able to recognize the gesture when the hand is displayed in the specified area. As the color pallet of the background is different from skin color pallet it shows no problem in responding according to the hand postures. So, in terms of simple/non-complex background this program shows great results.

Regarding Complex background, it was observed that the program had few difficulties in tracking the hand. Though with low accuracy rate the program was able to recognize the hand postures and responded accordingly.

The experiment shows promising result in both simple and complex background. Though it showed lower accuracy rate in complex backgrounds then the simple backgrounds it was still higher than the program with tracking based on complex back ground.

**Table 4-B:** Performance comparison (OpenCV with NumPy)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Background Type | Accuracy | Error | Tracking Rate | System |
| Simple (Non-Complex) | 95% | 5% | None | Core i5 CPU with 2.2 GHz processor |
| Complex | 70% | 30% |

In Table 4-B we illustrated what we have observed from the performance of the program. The program was run in Core i5 CPU with 2.2 GHz processor system. We operated the program in two different types of background. One was Simple background which did not have any large mix of color pallet and also did not match with skin color pallet. In this case we used white wall. Another was Complex background which had large mixture of color pallets and/or matched with skin color pallet like doors or windows. For each background we have tried gestures 20 times. And as mentioned before we raised our fingers from one to five in both hand and keep the hand steady for 5 seconds each. For Simple background the program was able to recognize the gestures successfully 19 times and for Complex background it successfully recognized 14 times. But unlike the other program this program did not have the tracking feature. So, there is no tracking rate for this program.

Our first way to recognize the hand gesture is a tracking-based method. In this method wherever the hand is it can be extracted from the background. In Table 2, the method got 90% accuracy and 10% of error in simple background. Complex background reduces the accuracy to 20% and as produce an error rate of 80%. On the other hand, from Table 3, another method that is without tracking produces a 95% accuracy in simple background with 5% of error and in complex background produces 70% of accuracy including 30% of error rate. It can be seen that both the method works well in simple background but when it comes to complex background, in both case of with and without tracking a dramatical change arise. In non-tracking method the outcome is quite desirable but the main problem of the method is the hand must be in a pre-defined position to be tracked which is quite unexpected during teaching session. So, if the hand is not in the expected position the recognition will fall as well as the whole system. Now come to the tracking method, though the method is not producing a good result but its main advantage is that the hand can be recognize in any position. So, what is necessary to change factors like light, color range and others to be modified to gain optimum result with the tracking method.

**4.2.3 Comparison of Experimental Results with Previous Research Results**

**Table 4-C:** Performance comparison between modified method and previous method

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Method | Accuracy / Recognition Rate | Error Rate | Processing Time (In Second) | Name of Background | Processing Speed in Frame Per Second |
| The Extrusion Technique [16] | 80% | 20% | The AR application took 53.8 s  GeoGebra took 71.95 s  Cabri3D took 47.67 s | Not mentioned | (depends on processor) |
| Modified extrusion technique (Tracking) | 90% | 10% | The AR application took 9 s to recognize the hand gesture. | Simple | The process used 25% of processor  in intel corei5 7th generation laptop. |
| 20% | 80% | Complex |
| Modified extrusion technique (Non-Tracking) | 95% | 5% | Simple |
| 70% | 30% | Complex |

**Fig. 4-2:** Accuracy among the modified techniques and previous technique

**Fig. 4-3:** Error among the modified techniques and previous technique

## 4.3 Conclusion

## It is clearly noticeable that the extrusion technique has got the highest position with the accuracy rate of 80%. The background was not disclosed in the research. Our proposed modified extrusion technique without tracking got the second place with 95% of recognition of accuracy rate when it was simple background. The recognition rate become 70% when the background become complex. Dramatically the accuracy rate falls to 20% when we applied modified extrusion technique with tracking. On the contrary it was 90% when the background was simple. Opposite scenario was seen in terms of error rate. As the recognition rate is higher in the extrusion technique, the error rate is lower. Non-tracking modified extrusion technique got 5% and 30% error rate in simple and complex background respectively. 10% error occurred with modified extrusion technique with tracking, but the amount rises when the background became complex and the rate was 80%. The extrusion technique took 58.3 s to process the whole system. But we only deal with the recognition of hand, not the whole system. Until now our system took 9s to complete its recognition task. Extrusion technique can be called a process dependent technique as the processing speed depends on the processor.

**Chapter 5: Conclusion and Future Work**

## 5.1 Introduction

The paper presents an approach for using hand-recognition systems based on the processing of an incoming digital image for manipulation in Augmented reality for 3D manifestation and control. The main objective of this research is to develop a system where an Augmented reality tool will be controlled using hand gesture recognition. As mentioned above we are following “The Modified Extrusion Technique” method which is an abridged version of “The Extrusion Technique” method. In regard of the Gesture Processing and Action mapping steps, we have implemented two hand gesture recognition application thus far. One is with tracking of hand and the other is without tracking of hand. We have also documented user experiments on the detection accuracy of the implemented application.

## 5.2 Contribution

As previously mentioned, we have completed following “The Modified Extrusion Technique” method to part of Action mapping which concludes the hand gesture recognition. In the original “The Modified Extrusion Technique” method it used a leap motion device that connects with a computer and enables users to manipulate digital objects with hand motions. This leap motion device is both costly and hard to find in some region as it is a newly developed technology. So instead of using a leap motion device for hand motion detection we came to a decision to use hand gesture recognition technology via camera which is less costly and easy to find. So, for that purpose we have implemented two hand gesture recognition application for experimental purpose where one first tracks the hand then recognizes the gesture and the other doesn’t need to track the hand and just recognizes the posture. Even though they are different when it comes to tracking, they work the same way when recognizing the hand gesture.

First of all, the webcam captures real time hand movement images as input for the applications. The applications then process those incoming digital images in real time and separate the image of a hand from the background. In this case we have found in our research that with tracking the hand from vast area gives less efficient result in complex background. But recognizing the hand from a specified position rather than to first track the hand gives more promising results. Secondly, the application would then perform binary normalization and then morphological image extraction. After successful recognition of the hand the applications give results according to the postures. Various types of posture are enlisted for recognition, but we only worked with the numbers of fingers to testify whether the applications can recognize hand or not. In order to gain more efficient results, the applications need further improvement and experimentation.

## 5.3 Future Work

As future work, we plan to use the applications outputs as inputs for the Augmented reality tool. To begin, we would start using the application with tracking with the tool in order to observe how proficiently the outputs of that application work with the tool. Same goes for the second application which is without tracking.

Even though for research purposes we will be experimenting with both applications we plan to give more priority to the application without hand tracking as it bodes well with our objective and also have higher accuracy rate.

In first display the user would be able to visual a 3D geometric object in the screen. In order to control that object user would need to synchronize its hands position with the object. Then the system would start processing images from the hand gesture the display results accordingly. This synopsis does not require the hand gesture recognition application to track the hand from any position. Rather it would only need to scan the hand when it come to the area of the 3D geometric object. This is why the second application takes more priority than the first.

After successful completion of the system we would survey the system among students for suggestions and improvement.

Furthermore, we plan to create individual devices for this system rather than using Computer as an only platform. We plan to create this device in a way that it would be accessible any time anywhere.

Apart from that we also plan to use the system for not only visualizing geometrical 3D objects but also objects of other study materials or any kind. People would be able to use it for not only teaching purpose but also any kind of work in general.

## 5.4 Conclusion

## The proposed system will play vital role in our education system. Augmented reality can immensely offer interactive learning experiences. And with the help of Hand recognition technology it will become a generation of touchless technology. As we told before it started with mathematics and in near future if the field of augmented reality can be nourished, other parts of education like medical science, chemical science, astrology all can be converted with augmented reality and a new era of technology will be open.

**References**

[1] Kaufrmann, Hames and Schmalstieg, Dieter, “Mathematics and Geometry Education with Collaboration Augmented Reality”, Computers & Graphics, vol. 27, issue 3, June 2003, pages 339-345. Accessed at: https://doi.org/10.1016/S0097-8493 (03)00028-1.

[2] Azad et al. (2018), “Dynamic 3D hand gesture recognition by learning weighted Depth motions”, IEEE Transactions on Circuits and Systems for Video Technology, doi:10.1109/TCSVT.2018.2855416, Accessed at: https://ieeexplore.ieee.org/document/8410578.

[3] Chen et al. (2017), “A hybrid CNN-SVM classifier for hand gesture recognition with surface EMG signals”, 2018 International Conference on Machine Learning and Cybernetics doi:10.1109/ICMLC.2018.8526976.

[4] Jung et al. (2017), “Boosthand: Distance-free object manipulation system with switchable non-linear mapping for augmented reality classrooms”, [2017 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)](https://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=8087877). doi: 10.1109/ISMAR-Adjunct.2017.96

[5] M.I.N.P. Munasinghev, “Dynamic hand gesture recognition using computer vision and neural networks”, 2018 3rd International Conference for Convergence in Technology (I2CT). doi: 10.1109/I2CT.2018.8529335

[6] Alam et al. (2018), “Improved gesture recognition using deep neural network on sEMG”, 2018 International Conference on Engineering, Applied Sciences, and Technology (ICEAST) doi:10.1109/ICEAST.2018.8434493.

[7] Alani et al. (2018), “Trajectory based hand gesture recognition using Kinect via deterministic learning”, 2018 37th Chinese Control Conference (CCC). doi:10.23919/ChiCC.2018.8482621

[8] Rani et al. (2017), “Hand gesture control of virtual object in augmented reality”, **2017** International Conference on Advances in Computing, Communications and Informatics (ICACCI). doi:10.1109/ICACCI.2017.8126053.

[9] Krupaka et al. (2017), “Toward realistic hand gesture interface: keeping it simple for developers and machine”.

CHI '17 Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems Pages 1887-1898.  
doi:10.1145/3025453.3025508

[10] Chiang et al. (2017). 3D depth information based 2D low complexity hand posture and gesture recognition design for human computer interaction. Accessed at: <https://ieeexplore.ieee.org/abstract> /document/8463327, November, 2018.

[11] Krittameth et al. (2018), “Hand Gesture Recognition with inertial sensors”, 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). doi:10.1109/EMBC.2018.8513098

[12] Kui Liu (2014), “Fusion of inertial and depth sensor data for Robust hand gesture recognition”,IEEE Sensors Journal ( Volume: 14 , Issue: 6 , June 2014 ).

doi:10.1109/JSEN.2014.2306094

[13] Liu et al. (2014), “Multi HMM classification for hand gesture recognition using two different modality sensors”, 2014 IEEE Dallas Circuits and Systems Conference (DCAS). doi:10.1109/DCAS.2014.6965338.

[14] Gudavalli et al. (2018), “Hand gesture recognition based on cascading of multiple feature”. 2018 International Conference on Intelligent Autonomous Systems (ICoIAS). doi:10.1109/ICoIAS.2018.8493816.

[15] Oka et al. (2002), “Real-time fingertips tracking and gesture recognition”,**IEEE** Computer Graphics and Applications (Volume: 22, Issue: 6 , Nov/Dec 2002 ) .**doi:**10.1109/MCG.2002.1046630

[16]. Hong-Quan Le, Jee-In Kim,“An Augmented Reality Application with Hand Gestures for Learning 3D Geometry”. 2017 IEEE International Conference on Big Data and Smart Computing (BigComp). doi:10.1109/BIGCOMP.2017.7881712

[17]. Meiping Tao, Li Ma, “A Hand Gesture Recognition Model Based on Semi-supervised Learning”, 2015 7th International Conference on Intelligent Human-Machine Systems and Cybernetics, doi:10.1109/IHMSC.2015.230

[18].DeviraAnggi Maharani, HanifFakhrurroja, Riyanto, CarmadiMachbub, “Hand Gesture Recognition Using K-Means Clustering and Support Vector Machine”, **2018** IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE), doi:10.1109/ISCAIE.2018.8405435.

[19]. D.K. Vishwakarma, Sahib Majithia, [Nikhil Kumar Mishra](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=%22First%20Name%22:%22Nikhil%20Kumar%22&searchWithin=%22Last%20Name%22:%22Mishra%22&newsearch=true&sortType=newest), “A Static Hand Gesture Recognition System to Recognize the Total number of Fingers”, 2017 International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE). doi:10.1109/RISE.2017.8378176

[20] TanatchaChaikhumpha,PhattanaphongChomphuwiset , “Real – time Two Hand Gesture Recognition with Condensation and Hidden Markov Models”, 2018 International Workshop on Advanced Image Technology (IWAIT). doi:10.1109/IWAIT.2018.8369811

[21] Fan Zhang, Yue Liu, ChunyuZou, Yongtian Wang, “Hand gesture recognition based on HOG-LBP feature”, 2018 IEEE International Instrumentation and Measurement Technology Conference (I2MTC). **doi:**10.1109/I2MTC.2018.8409816

[22]. Ali A. Alani, Georgina Cosma, AboozarTaherkhani ,T.M McGinnity,“Hand Gesture Recognition Using an Adapted Convolutional Neural Network with Data Augmentation”.2018 4th International Conference on Information Management (ICIM). doi:10.1109/INFOMAN.2018.8392660

[23]. [Dinesh Kumar Vishwakarma](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=%22First%20Name%22:%22Dinesh%20Kumar%22&searchWithin=%22Last%20Name%22:%22Vishwakarma%22&newsearch=true&sortType=newest),“Hand Gesture Recognition using Shape and Texture evidences in Complex Background”, 2017 International Conference on Inventive Computing and Informatics (ICICI). doi:10.1109/ICICI.2017.8365354

[24] P. N. V. S. Gowtham, “A Hand Gesture Recognition Based Virtual Touch Wall”, IJIET 2012 Vol.2 (1): 36-42 ISSN: 2010-3689. doi:10.7763/IJIET. 2012.V2.7

[25]. Pichao Wang1, Wanqing Li1, Song Liu1, Zhimin Gao1, Chang Tang2 and Philip Ogunbona1 “Large-scale Isolated Gesture Recognition Using Convolutional Neural Networks”, 2016 23rd International Conference on Pattern Recognition (ICPR). doi:10.1109/ICPR.2016.7899599

[26]. M. Yeasin, S. Chaudhuri, “Dynamic hand gesture understanding - a new approach”, 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No.99CH36258). doi:10.1109/ICASSP.1999.757490

[27].Yunan Li , Qiguang Miao, KuanTian, Yingying Fan, XinXu, RuiLi,Jianfeng Song , “Large-scale Gesture Recognition with a Fusion of RGB-D Data Based on the C3D model”, 2016 23rd International Conference on Pattern Recognition (ICPR), doi:10.1109/ICPR.2016.7899602

[28]. Ryan Christopher Yeoh,StevenZhiYing Zhou , “Consistent Real-time Lighting for Virtual Objects in Augmented Reality”. 2009 8th IEEE International Symposium on Mixed and Augmented Reality, doi:10.1109/ISMAR.2009.5336453

[29]. HartmutSeichter, “SKETCHAND+ a collaborative augmented reality sketching application”,Accessed at: <https://www.researchgate.net/publication/30874877_Sketchand_a_Collaborative_Augmented_Reality_Sketching_Application>

[30]. Tiffany Leung, FarhanaZulkernine, HarunaIsah, “The use of Virtual Reality in Enhancing Interdisciplinary Research and Education”, Accessed at: https://arxiv.org/abs/1809.08585

[31]. A. Butz T. Hollerer,S.Feiner, B. MacIntyre,C. Beshers , “Enveloping Users and Computers in a Collaborative 3D Augmented Reality”, Proceedings 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99), doi:10.1109/IWAR.1999.803804

[32]. James Purnama, Daniel Andrew, MaulahikmahGalinium , “Geometry Learning Tool for Elementary School using Augmented Reality”, 2014 International Conference on Industrial Automation, Information and Communications Technology, doi:10.1109/IAICT.2014.6922112

[33]. H. Kaufmann, D. Schmalstieg, “Designing Immersive Virtual Reality for Geometry Education”, IEEE Virtual Reality Conference (VR 2006), doi:10.1109/VR.2006.48

[34]. [Hannes Kaufmann](https://www.researchgate.net/profile/Hannes_Kaufmann), Karin Steinbügl, “Improving Spatial Abilities by Geometry Education in Augmented Reality - Application and Evaluation Design”, Accessed at: https://www.researchgate.net/publication/216867616\_Improving\_Spatial\_Abilities\_by\_Geometry\_Education\_in\_Augmented\_Reality\_Application\_and\_Evaluation\_Design\_Proceedings.

[35]. Hannes Kaufmann, “The potential of augmented reality in dynamic geometry education”, Accessed at: https://www.researchgate.net/publication/228350641\_The\_potential\_of\_augmented\_reality\_in\_dynamic\_geometry\_education.

[36].HannesKaufmann, DieterSchmalstieg, “Mathematics and Geometry Education with Collaborative Augmented Reality”, Accessed at: https://doi.org/10.1016/S0097-8493 (03)00028-1

[38] Manavender R. Malgireddy, “A Framework for Hand Gesture Recognition and Spotting Using Sub-Gesture Modeling”, https://ieeexplore.ieee.org/document/5597566/

[39] Fan Zhang, “Hand gesture recognition based on HOG-LBP feature”,2018 IEEE International Instrumentation and Measurement Technology Conference (I2MTC),doi: 10.1109/I2MTC.2018.8409816

[40] Dinesh Kumar Vishwakarma, Hand Gesture Recognition in Low-Intensity Environment Using Depth Images [2017 International Conference on Intelligent Sustainable Systems (ICISS)](https://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=8376163),doi:10.1109/ISS1.2017.8389446

[41]NUSHandPostureDataset,https://www.ece.nus.edu.sg/stfpa ge/elepv/NUS-HandSet. University Science, 1989.

[42]Barczak, A.L.C.,Reyes, N.H.,Abastillas, M.,Piccio, A.,Susnjak, T.,“A new 2D static hand gesture colour image dataset for ASL gestures”, Massey University, Accessed at : http://hdl.handle.net/10179/4514

[43] MarimpisAvraam, “Static Gesture Recognition Combining Graph and Appearance Features’’, International Journal of Advanced Research in Artificial Intelligence (IJARAI), Volume 3 Issue 2, 2014. doi: 10.14569/IJARAI.2014.030201

[44] Jaehong Lee,Heon Gu,Hyungchan Kim,Jungmin Kim,Hyoungrae Kim,Hakil Kim, “Interactive manipulation of 3D objects using Kinect for visualization tools in education”, 2013 13th International Conference on Control, Automation and Systems (ICCAS 2013), doi: 10.1109/ICCAS.2013.6704175

[45] Rajat Shrivastava, "A hidden Markov model based dynamic hand gesture recognition system using OpenCV", 2013 3rd IEEE International Advance Computing Conference (IACC), doi: 10.1109/IAdCC.2013.6514354

[46] Stefan van der Walt , S. Chris Colbert , Gael Varoquaux, "The NumPy Array: A Structure for Efficient Numerical Computation", Computing in Science & Engineering ( Volume: 13 , Issue: 2 , March-April 2011 ), doi: 10.1109/MCSE.2011.37

**Appendix A**

Code of gesture recognition using EmguCV:

**CustomYCrCbSkinDetector.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using Emgu.CV;

using Emgu.CV.Structure;

namespace HandGestureRecognition.SkinDetector

{

class CustomYCrCbSkinDetector:IColorSkinDetector

{

public override Image<Gray, byte> DetectSkin(Image<Bgr, byte> Img, IColor min, IColor max)

{

//Code adapted from here

// http://blog.csdn.net/scyscyao/archive/2010/04/09/5468577.aspx

// Look at this paper for reference (Chinese!!!!!)

// http://www.chinamca.com/UploadFile/200642991948257.pdf

Image<Ycc,Byte> currentYCrCbFrame = Img.Convert<Ycc, Byte>();

Image<Gray, Byte> skin = new Image<Gray, Byte>(Img.Width, Img.Height);

int y, cr, cb, l, x1, y1, value;

int rows = Img.Rows;

int cols = Img.Cols;

Byte[, ,] YCrCbData = currentYCrCbFrame.Data;

Byte[, ,] skinData = skin.Data;

for (int i = 0; i < rows; i++)

for (int j = 0; j < cols; j++)

{

y = YCrCbData[i, j, 0];

cr = YCrCbData[i, j, 1];

cb = YCrCbData[i, j, 2];

cb -= 109;

cr -= 152;

x1 = (819 \* cr - 614 \* cb) / 32 + 51;

y1 = (819 \* cr + 614 \* cb) / 32 + 77;

x1 = x1 \* 41 / 1024;

y1 = y1 \* 73 / 1024;

value = x1 \* x1 + y1 \* y1;

if (y < 100)

skinData[i, j, 0] = (value < 700) ? (byte)255 : (byte)0;

else

skinData[i, j, 0] = (value < 850) ? (byte)255 : (byte)0;

}

StructuringElementEx rect\_6 = new StructuringElementEx(6, 6, 3, 3, Emgu.CV.CvEnum.CV\_ELEMENT\_SHAPE.CV\_SHAPE\_RECT);

CvInvoke.cvErode(skin, skin, rect\_6, 1);

CvInvoke.cvDilate(skin, skin, rect\_6, 2);

return skin;

}

}

}

**HsvSkinDetector.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using Emgu.CV;

using Emgu.CV.Structure;

namespace HandGestureRecognition.SkinDetector

{

public class HsvSkinDetector:IColorSkinDetector

{

public override Image<Gray, byte> DetectSkin(Image<Bgr, byte> Img, IColor min, IColor max)

{

Image<Hsv, Byte> currentHsvFrame = Img.Convert<Hsv, Byte>();

Image<Gray, byte> skin = new Image<Gray, byte>(Img.Width, Img.Height);

skin = currentHsvFrame.InRange((Hsv)min,(Hsv)max);

return skin;

}

}

}

IColorSkinDetector.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using Emgu.CV;

using Emgu.CV.Structure;

namespace HandGestureRecognition.SkinDetector

{

public abstract class IColorSkinDetector

{

public abstract Image<Gray, Byte> DetectSkin(Image<Bgr, Byte> Img, IColor min, IColor max);

}

}

YCrCbSkinDetector.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using Emgu.CV.Structure;

using Emgu.CV;

namespace HandGestureRecognition.SkinDetector

{

public class YCrCbSkinDetector:IColorSkinDetector

{

public override Image<Gray, byte> DetectSkin(Image<Bgr, byte> Img, IColor min, IColor max)

{

Image<Ycc, Byte> currentYCrCbFrame = Img.Convert<Ycc, Byte>();

Image<Gray, byte> skin = new Image<Gray, byte>(Img.Width, Img.Height);

skin = currentYCrCbFrame.InRange((Ycc)min,(Ycc) max);

StructuringElementEx rect\_12 = new StructuringElementEx(12, 12, 6, 6, Emgu.CV.CvEnum.CV\_ELEMENT\_SHAPE.CV\_SHAPE\_RECT);

CvInvoke.cvErode(skin, skin, rect\_12, 1);

StructuringElementEx rect\_6 = new StructuringElementEx(6, 6, 3, 3, Emgu.CV.CvEnum.CV\_ELEMENT\_SHAPE.CV\_SHAPE\_RECT);

CvInvoke.cvDilate(skin, skin, rect\_6, 2);

return skin;

}

}

}

Run from this solution

**Program.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Windows.Forms;

namespace HandGestureRecognition

{

static class Program

{

/// <summary>

/// The main entry point for the application.

/// </summary>

[STAThread]

static void Main()

{

Application.EnableVisualStyles();

Application.SetCompatibleTextRenderingDefault(false);

Application.Run(new Form1());

}

}

}

**Code of gesture recognition using NumPy:**

import cv2

import numpy as np

import math

cap = cv2.VideoCapture(0)

while cap.isOpened():

try: # an error comes if it does not find anything in window as it cannot find contour of max area

# therefore this try error statement

ret, frame = cap.read()

frame = cv2.flip(frame, 1)

kernel = np.ones((3, 3), np.uint8)

# define region of interest

roi = frame[100:300, 100:300]

cv2.rectangle(frame, (100, 100), (300, 300), (0, 255, 0), 0)

hsv = cv2.cvtColor(roi, cv2.COLOR\_BGR2HSV)

# define range of skin color(BGR) in HSV

lower\_skin = np.array([0, 20, 70], dtype=np.uint8)

upper\_skin = np.array([20, 255, 255], dtype=np.uint8)

# extract skin colur imagw

# Threshold the HSV image to get only skin colors

mask = cv2.inRange(hsv, lower\_skin, upper\_skin)

# extrapolate the hand to fill dark spots within

mask = cv2.dilate(mask, kernel, iterations=4)

# blur the image

mask = cv2.GaussianBlur(mask, (5, 5), 100)

# find contours

contours, hierarchy = cv2.findContours(mask, cv2.RETR\_TREE, cv2.CHAIN\_APPROX\_SIMPLE)

# find contour of max area(hand)

cnt = max(contours, key=lambda x: cv2.contourArea(x))

# approx the contour a little

epsilon = 0.0005 \* cv2.arcLength(cnt, True)

approx = cv2.approxPolyDP(cnt, epsilon, True)

# make convex hull around hand

hull = cv2.convexHull(cnt)

# define area of hull and area of hand

areahull = cv2.contourArea(hull)

areacnt = cv2.contourArea(cnt)

# find the percentage of area not covered by hand in convex hull

arearatio = ((areahull - areacnt) / areacnt) \* 100

# find the defects in convex hull with respect to hand

hull = cv2.convexHull(approx, returnPoints=False)

defects = cv2.convexityDefects(approx, hull)

# l = no. of defects

l = 0

# code for finding no. of defects due to fingers

for i in range(defects.shape[0]):

s, e, f, d = defects[i, 0]

start = tuple(approx[s][0])

end = tuple(approx[e][0])

far = tuple(approx[f][0])

pt = (100, 180)

# find length of all sides of triangle

a = math.sqrt((end[0] - start[0]) \*\* 2 + (end[1] - start[1]) \*\* 2)

b = math.sqrt((far[0] - start[0]) \*\* 2 + (far[1] - start[1]) \*\* 2)

c = math.sqrt((end[0] - far[0]) \*\* 2 + (end[1] - far[1]) \*\* 2)

s = (a + b + c) / 2

ar = math.sqrt(s \* (s - a) \* (s - b) \* (s - c))

# distance between point and convex hull

d = (2 \* ar) / a

# apply cosine rule here

angle = math.acos((b \*\* 2 + c \*\* 2 - a \*\* 2) / (2 \* b \* c)) \* 57

# ignore angles > 90 and ignore points very close to convex hull(they generally come due to noise)

if angle <= 90 and d > 30:

l += 1

cv2.circle(roi, far, 3, [255, 0, 0], -1)

# draw lines around hand

cv2.line(roi, start, end, [0, 255, 0], 2)

l += 1

# print corresponding gestures which are in their ranges

font = cv2.FONT\_HERSHEY\_SIMPLEX

if l == 1:

if areacnt < 2000:

cv2.putText(frame, 'Put hand in the box', (0, 50), font, 2, (0, 0, 255), 3, cv2.LINE\_AA)

else:

if arearatio < 12:

cv2.putText(frame, '0', (0, 50), font, 2, (0, 0, 255), 3, cv2.LINE\_AA)

elif arearatio < 17.5:

cv2.putText(frame, 'Best of luck AIUB', (0, 50), font, 2, (0, 0, 255), 3, cv2.LINE\_AA)

else:

cv2.putText(frame, '1', (0, 50), font, 2, (0, 0, 255), 3, cv2.LINE\_AA)

elif l == 2:

cv2.putText(frame, '2', (0, 50), font, 2, (0, 0, 255), 3, cv2.LINE\_AA)

elif l == 3:

if arearatio < 27:

cv2.putText(frame, '3', (0, 50), font, 2, (0, 0, 255), 3, cv2.LINE\_AA)

else:

cv2.putText(frame, 'ok', (0, 50), font, 2, (0, 0, 255), 3, cv2.LINE\_AA)

elif l == 4:

cv2.putText(frame, '4', (0, 50), font, 2, (0, 0, 255), 3, cv2.LINE\_AA)

elif l == 5:

cv2.putText(frame, '5', (0, 50), font, 2, (0, 0, 255), 3, cv2.LINE\_AA)

elif l == 6:

cv2.putText(frame, 'reposition', (0, 50), font, 2, (0, 0, 255), 3, cv2.LINE\_AA)

else:

cv2.putText(frame, 'reposition', (10, 50), font, 2, (0, 0, 255), 3, cv2.LINE\_AA)

# show the windows

except:

pass

cv2.imshow('mask', mask)

cv2.imshow('frame', frame)

k = cv2.waitKey(5) & 0xFF

if k == 27:

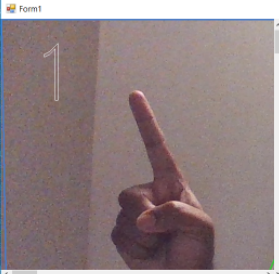
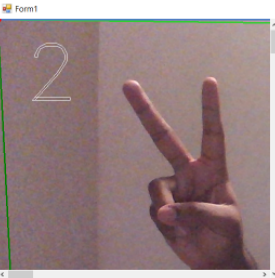
break

cv2.destroyAllWindows()

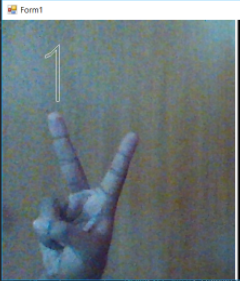
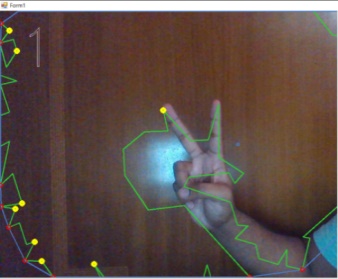
cap.release()

**Appendix B**

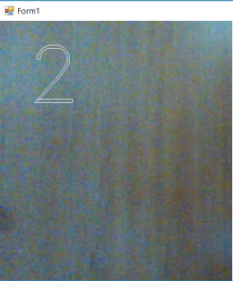
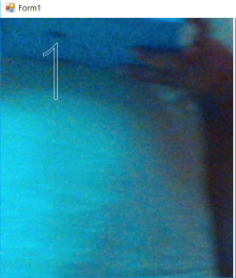
**Screenshot of gesture recognition using EmguCV**



Recognition in Simple Background

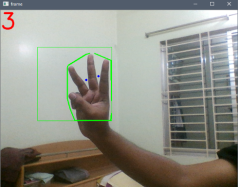
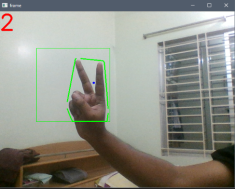


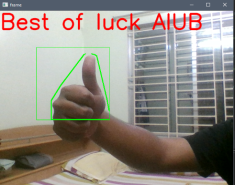
Recognition in Complex Background



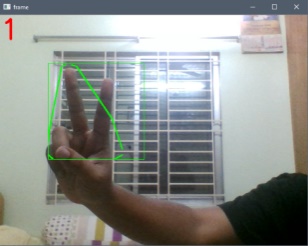
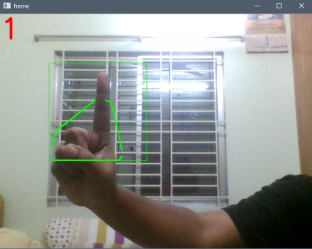
Incorrect Recognition in Complex Background

**Screenshot of gesture recognition using OpenCv with NumPy**





Recognition in Simple Background



Incorrect Recognition in Complex Background