2021

Automated Tracking of Hand Hygiene Stages

Rashmi Bakshi
Technological University Dublin

Automated Tracking of Hand Hygiene Stages

Rashmi Bakshi

Supervised by:

Dr Graham Gavin; Dr Jane Courtney; Dr Damon Berry

Thesis presented for the degree of

Master of Philosophy

School of Electrical and Electronic Engineering

TU Dublin

Ireland

2021

Abstract

The European Centre for Disease Prevention and Control (ECDC) estimates that 2.5 million cases of Hospital Acquired Infections (HAIs) occur each year in the European Union. Hand hygiene is regarded as one of the most important preventive measures for HAIs. If it is implemented properly, hand hygiene can reduce the risk of cross-transmission of an infection in the healthcare environment. Good hand hygiene is not only important for healthcare settings. The recent ongoing coronavirus pandemic has highlighted the importance of hand hygiene practices in our daily lives, with governments and health authorities around the world promoting good hand hygiene practices. The WHO has published guidelines of hand hygiene stages to promote good hand washing practices. A significant amount of existing research has focused on the problem of tracking hands to enable hand gesture recognition. In this work, gesture tracking devices and image processing are explored in the context of the hand washing environment. Hand washing videos of professional healthcare workers were carefully observed and analyzed in order to recognize hand features associated with hand hygiene stages that could be extracted automatically. Selected hand features such as palm shape (flat or curved); palm orientation (palms facing or not); hand trajectory (linear or circular movement) were then extracted and tracked with the help of a 3D gesture tracking device - the Leap Motion Controller. These features were further coupled together to detect the execution of a required WHO - hand hygiene stage, *Rub hands palm to palm*, with the help of the Leap sensor in real time. In certain conditions, the Leap Motion Controller enables a clear distinction to be made between the left and right hands. However, whenever the two hands came into contact with each other, sensor data from the Leap, such as palm position and palm orientation was lost for one of the two hands. Hand occlusion was found to be a major drawback with the application of the device to this use case. Therefore, RGB digital cameras were selected for further processing and tracking of the hands. An image processing technique, using a skin detection algorithm, was applied to extract instantaneous hand positions for further processing, to enable various hand hygiene poses to be detected. Contour and centroid detection algorithms were further applied to track the hand trajectory in hand hygiene video recordings. In addition, feature detection algorithms were applied to a hand hygiene pose to extract the useful hand features. The video recordings did not suffer from occlusion as is the case for the Leap sensor, but the segmentation of one hand from another was identified as a major challenge with images because the contour detection resulted in a continuous mass when the two hands were in contact. For future work, the data from gesture trackers, such as the Leap Motion Controller and cameras (with image processing) could be combined to make a robust hand hygiene gesture classification system.

As a part of the structured post-graduate programme, I have successfully completed all the modules accounting for 40 ECTS.

Modules attended were:

1. Research Methods
2. Social Network Analysis
3. User Experience Design
4. Introduction to Statistics
5. Research Integrity
6. Image Processing
7. Introduction to Pedagogy
8. Software Engineering

Publication from this Thesis

1. R.Bakshi, J. Courtney, D. Berry, and G. Gavin, "Identification of Unique Features and Exploration of Leap Motion Controller for Detecting Hand Hygiene Stages," 24th Annual Conference , Bio-engineering of the Royal Academy of Medicine, pp. 26–27, 2018.
2. R.Bakshi, J.Courtney, D.Berry and G. Gavin, "Suitability of Leap Motion Controller for detecting hand hygiene gestures", 25th Annual Conference, Bio-engineering of the Royal Academy of Medicine, 2019.
3. R.Bakshi, J.Courtney, D.Berry and G. Gavin, "Gesture Analysis: Hand Hygiene", INSPIRE Conference, TU-Dublin, 9th May, 2019.
4. R.Bakshi, J.Courtney, D.Berry and G.Gavin,"Tracking Hand Trajectory as a preliminary study for Hand Hygiene stages", Irish Machine Vision and Image Processing Conference, 2019.

Contents

Chapter 1

Introduction

Interaction design is the field of study that facilitates an interaction between a user and a system. A system can be a real-time mobile application, a responsive website or a virtual interface with an overall role of providing an intuitive digital and user interactive experience. Gestural interaction is beginning to become an essential part of modern-day interaction design where a user may interact with a system with the help of hand gestures [1]. Motion based game controllers such as the Microsoft Kinect allow for full body gesture interaction at an affordable price. Gesture interaction is gaining popularity with the increasing interactive demands of the gaming industry. However, besides the entertainment industry and other industries, such as manufacturing and automation, gesture interaction is expanding in the healthcare sector for providing assistance to patients with mobility and other health-related issues [2].

One example where the tracking and identification of gestures are of interest is the process of hand washing. This has come to the fore in recent times as a result of the Covid-19 pandemic and the seasonal flu, where good hand hygiene practices are known to mitigate the spread of these viruses. The process of hand washing involves dynamic hand gestures. It may be possible to analyse these hand gestures and extract unique hand features for detection and classification with use of motion-based game controllers, gesture trackers, and cameras. The prospect of detecting the stages involved in the process of hand washing may be useful in a healthcare setting where it is proven that correct hand hygiene practices can reduce the rate of hospital acquired infections [3].

1.1 A brief history of interactive interfaces

In 1968, Doughlas Engelbart presented a working prototype at the Fall Joint Computer Conference, demonstrating his vision of interactive computing including window display screens, mouse, hyper-text, and multimedia applications. This was the origin of interactive interfaces [4]. In 1982, the first human-controlled multi-touch device was developed at the University of Toronto by Nimish Mehta [5]. Around the same time, Bill Buxton, who played a major role in the multitouch technology, presented a multi touch tablet prototype which was capable of detecting multiple points of contact by using a capacitive grid [5]. Shortly, thereafter, Myron Kruger, an American computer artist introduced gestural interaction by developing an optical system that could track hand movements. *video place–* a telecommunication environment was developed that used projectors and video cameras to track hands, fingers and geographically separate participants. The work was not touch based but was advanced for the time and Kruger went on to contribute in areas such as virtual reality and in interaction design later in his career [6]. Touch screens became a part of daily life in the form of Personal Digital Assistance (PDAs) in 1990s. In 1992, IBM introduced the Simon Personal Communicator, one of the first mobile phones with touchscreen technology. It consisted of interactive applications such as email, address books, calendar, etc. It used a pen-based sketchpad and stylus to navigate through menus [7]. In the period of 10 years starting from the year 2000, major developments happened in gesture based interaction. In 2001 , 'Alias Wave Front' built a gesture-based interface, Portfolio Wall, that allowed users to interact with the images, videos, and 3D files with just their fingers [8]. In 2002, Sony introduced a flat input surface that could recognize multiple hand positions and touch points simultaneously. It was called 'Smart Skin'. The technology worked by calculating the distance between the hand and the surface with capacitive sensing and a mesh-shaped antenna. In contrast to camera-based gesture recognition systems, all sensing elements were integrated within the touch surface and this method did not suffer from poor lighting and occlusion problems [9]. In 2010, Microsoft launched the low cost depth camera, Microsoft Kinect; the sensor provides full body 3D motion capture, facial recognition and voice recognition. The depth sensor consists of an infrared projector and a camera for tracking purposes [10].

Most of these advanced interfaces allow human-computer interaction without physical contact but utilize specific hardware devices such as hand gloves or tracking devices for human motion capture [11].

1.2 The role of gesture trackers in health care

Gesture and motion tracking systems are gaining acceptance in the healthcare industry. Examples include providing necessary help to vulnerable and elderly patients.

Gestural interactions provide new opportunities to help elderly people in their mobility.

Understanding of human movements, behaviour and body language is becoming an essential element of ambient assisted living for developing prevention and monitoring applications in the healthcare sector [2].

Recognition of human movements is helpful in providing assistance with physical and cognitive exercises for the elderly [2]. SensHand- a device developed by Cavallo et al. composed of four sensor units, three to be worn on the hand as rings for detecting finger movements and one to be worn as a bracelet for wrist detection; is used for assessing motor skills in patients with Parkinson disease [2]. These types of wearable sensors need to be worn by the user which can feel uncomfortable and can limit the ability to perform gestures in a natural way. Yet, these sensors are used as they receive data directly from the user movement and overcome the limitation of obstruction and light [2].

Gesture tracking devices such as the Microsoft Kinect; shown in Figure 1.2.1 are used in developing a healthcare assistance system that consists of a robotic application for indoor and outdoor navigation [12]. They have been used in building a robotic arm that can imitate and perform functions of a human arm and can be useful in computer assisted surgery [13, 14]. They are also used in the classification of sign language gestures [15].

3D gesture trackers are cost effective and safe to use and thus they have received a growing interest in the health care imaging industry. Recent applications involve health monitoring, screening, rehabilitation, assistance systems, and intervention support [16]. Due to unobtrusive and accurate measurements, these sensors are gaining popularity in the early diagnosis of Parkinson's disease [17]. With help of these sensors, gesture-based applications are developed to track hand movements of the patients with impaired dexterity [18]. Wile et al. utilised smartwatch device to identify tremor in the hands and monitor Parkinson disease among patients [19].

Gesture tracking sensors have an immense potential and can be explored to develop different applications such as monitoring daily routine gestures of elderly people to reduce the risk for accidents occurring at home.

Figure 1.2.1: Hand gesture tracking device-Microsoft Kinect

1.3 Hand washing - a gesture recognition application

Washing hands is a part of our daily lives. It is the act of cleaning hands by removing soil, dirt and microorganisms. The process of hand washing includes dynamic and complex hand movements which involve the use of two hands. These complex hand movements can be further classified into dynamic traceable gestures [20].

Hand washing is a well-documented and structured process as per World Health Organisation guidelines. Automatic detection of hand hygiene stages and providing feedback to the user may prove to be beneficial with the goal of improving hand hygiene practices and compliance rates and reducing the risk of spread of hospital acquired infections. This leads us to investigate and understand the process of hand washing in detail, outlined below and also in the accompanying figure.

Research Question: Can 3D gesture tracking systems and cameras be used to track hand hygiene stages?

Research Objectives

1. Analysis of the process of hand washing in order to decompose the hand movements and extract the unique hand features associated with each hand hygiene stage.

2. Evaluate the suitability of a commercially available gesture tracker, the Leap Motion Controller by tracking hand features and detection of the hand hygiene stages.

3. Utilise camera images and video recordings by tracking various hand poses and movements involved in hand washing.

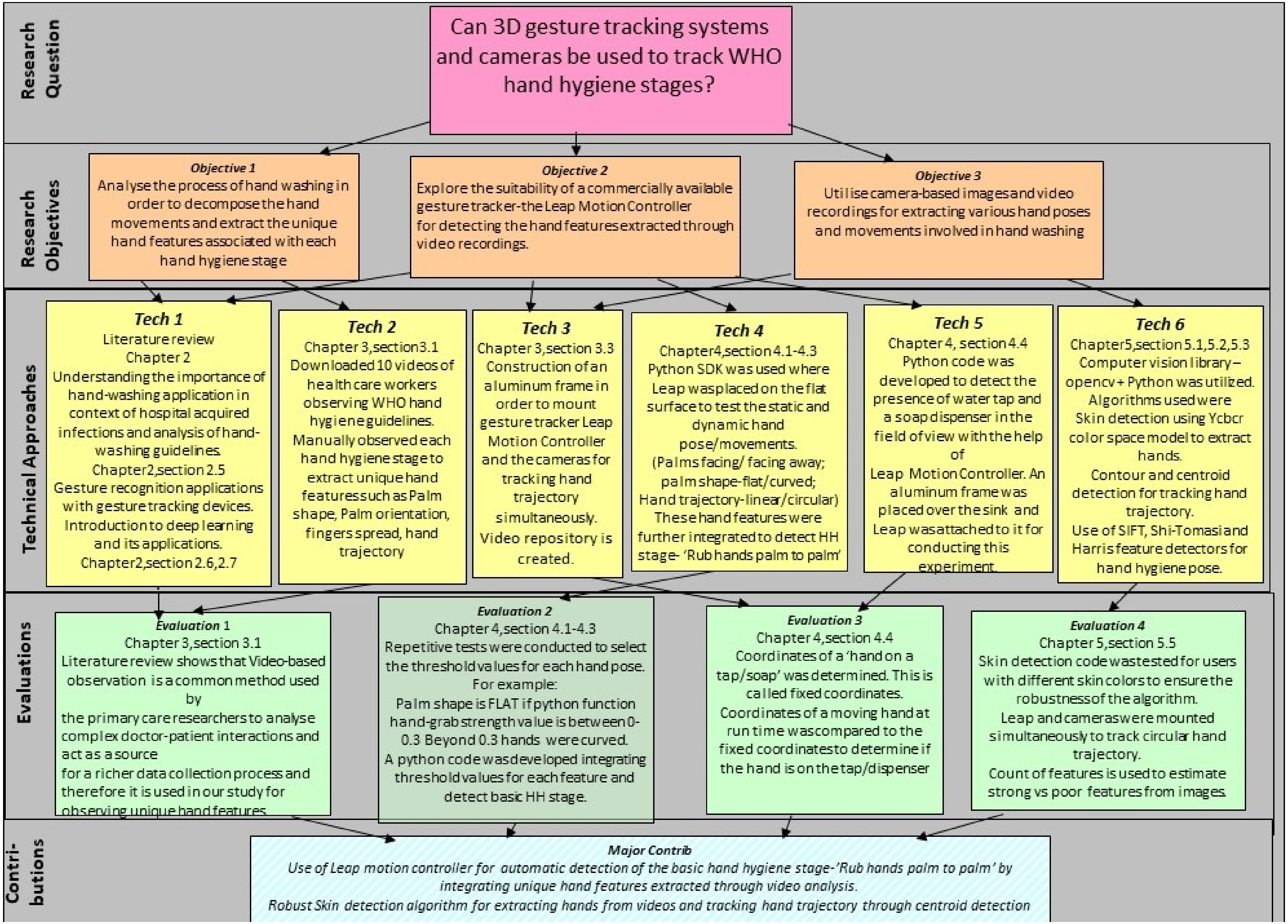

Figure 1.3.1: Overview of the Research Project.

1.4 Overview of the proposed work

This chapter has described the role of gesture tracking devices in the healthcare sector and has also provided a rationale for the development of a real-time hand washing application. It has also listed the research objectives that are planned for this work.

Chapter 2 describes the literature review for the spread of hospital acquired infections; the introduction about novel Covid-19; low compliance for hand hygiene practice; and existing hand gesture recognition systems with 3D sensors and cameras. The deep learning technology as a future direction of the work is discussed as well.

Chapter 3 describes the proposed methodology for unique hand feature extraction from hand washing videos and construction of an aluminium fixture rig to be placed over the laboratory's sink for the mounting of the cameras and the Leap Motion Controller. The hand washing video repository prepared for this work is discussed as well.

Chapter 4 describes the technical work carried out to evaluate the suitability of the Leap Motion Controller for tracking hand hygiene stages.

Chapter 5 describes the methodology and technical work for the use of cameras for tracking hands in hand hygiene video recordings.

Chapter 6 describes the conclusions of the work conducted as well as the fusion of the sensor Leap Motion Controller and camera systems for building of a robust hand gesture recognition system.

Chapter 2

Literature Review

2.1 Hospital acquired infections and the role of hand hygiene

Hospital acquired infections (HAIs) are the infections that are acquired in a health care setting through the direct contact with an infected worker, through patient-to-patient contact, or contact with contaminated equipment. They are unrelated to the original illness that brought the patient to the hospital [21]. MRSA-methicillin resistant staphylococcus aureus is one of the most common drug-resistant bacteria and a major cause for the spread of HAIs [22].

According to the European Centre for Disease Prevention and Control (ECDC), 2.5 million cases of HAIs occur in European Union and European Economic Area (EU/EAA) each year, corresponding to 2.5 million DALYs (disability adjusted life year) which is a measure of the number of years lost due to ill health, disability or an early death [23]. The spread of HAIs is a serious problem and needs to be addressed in an effective manner to decrease the number of infected patients.

A well-known method to reduce the rate of HAIs is through regular structured washing of hands. Healthcare authorities such as the World Health Organization (WHO), the Centres for Disease Control and Prevention (CDC) have published structured guidelines for practicing hand hygiene in health care settings. There is a standard approach for washing hands that should be adhered to by health care workers such as doctors, nurses, and physicians [24].

Hand hygiene is identified as a measure to prevent the cross-transmission of microorganisms and to reduce the rate of HAIs [3].

Hand hygiene practice can be classified into two categories:

- Inherent hand hygiene practice: where hands are visibly stained and/ or feel dirty and it acts as a trigger to wash hands thoroughly.

- Elective hand hygiene practice: such as social interactions while touching the patient, taking the pulse, lifting the patient, or recording the blood pressure. These behaviours do not trigger an intrinsic need to wash hands but they do involve the risk of cross transmission [25, 22].

Hand hygiene is a fundamental intervention in the control of HAIs transmitted by contact with the staff [26]. Song et al. conducted a study where they found that an increase in the consumption of alcohol-based hand rub (AHR) reduced the rate of MRSA infection. The amount of AHR was in the range between 3 and 78 mL/patient-day (pd) at the beginning and increased to 12 to 103 mL/pd at the end of the intervention studies. The hand hygiene compliance with AHR was in the range between 20% and 64% at the beginning and between 42% and 71% at the end of the study [27].
Washing hands is a simple, yet effective measure to prevent the spread of bacterial infections. The next section discusses the emergence of a novel Covid-19 in 2019 and the significance of regular hand hygiene practices to control the pandemic.

2.2 Novel corona virus disease Covid-19

The novel corona virus disease (Covid-19) is caused by the virus SARS-CoV-2. The inception was in Wuhan, China, but the virus has spread worldwide, with over 101,700 cases and 3,461 deaths in more than 75 countries as of 14 March 2020 [28]. According to Covid-19 weekly epidemiological update, over 2.6 million new cases were reported in March 2021 [29].
Guan et al. extracted data of 1099 patients that tested positive for COVID-19 from different hospitals in 30 provinces. The most common symptoms were:

1. Fever (was observed in 43.8% patients on admission but developed in 88.7% patients during hospitalisation)

2. Cough(67.8% patients)

3. Nausea/vomiting (5%)

4. Diarrhoea(3.8%)-the least common symptom [30]

2.2.1 Transmission and spread

The first cases of COVID-19 disease, which were connected with direct exposure to the huanan seafood wholesale market of Wuhan, and therefore the animal-human transmission was presumed as the main cause of spread. Due to the lack of further cases for animal-human transmission, it was concluded that the virus could be transmitted from human-to-human contact. The data suggested that the use of isolation is the best way to contain the epidemic.
Based on the data from the first case in Wuhan, this epidemic doubled about every seven days where each patient transmits the infection to an additional 2.2 individuals on an average [31].

2.2.2 Treatment and prevention measures

There were no specific antiviral treatments for COVID-19 and vaccines are currently only in the development stage in the year 2020. Oxygen therapy and mechanical ventilation were provided in extreme cases of respiratory failure.
Preventive measures focus on limiting the spread of the disease. WHO and other health authorities have issued the following general guidelines to control the spread of this epidemic.

- Avoid close contact with subjects suffering from acute respiratory infections.
- Wash your hands frequently, especially after contact with infected people or their environment.
- Avoid unprotected contact with farm or wild animals.
- People with symptoms of acute airway infection should keep their distance, cover coughs or sneeze with disposable tissues or clothes and wash their hands.
- Strengthen, in particular, in emergency medicine departments, the application of strict hygiene measures for the prevention and control of infections.
- Individuals that are immune compromised should avoid public gatherings.

WHO places emphasis on frequent hand washing, the use of portable hand sanitisers and avoiding contact with the face and mouth after interacting with the contaminated environment [31].
As of January 2021, WHO has issued an emergency use of approved vaccines such as Pfizer COVID-19 vaccine and AstraZeneca/Oxford Covid-19 vaccine to control the spread of the disease.

2.2.3 Worldwide cases

The corona virus outbreak is an ongoing pandemic where the infected number of patients are changing daily. According to a World Health Organisation situation report published on 19 May 2020, 4,731,458 cases have occurred globally with 316,169 deaths.
The majority of the patients were infected through local transmission where the source of an infection was within the reporting location. “Imported cases” was another cause of the spread where the infection where the virus was acquired outside the location of reporting. “Community transmission” refers to the inability to relate confirmed cases through chains of transmission for a large number of cases or by increasing positive tests through sentinel samples (routine systematic testing of respiratory samples from established laboratories) [32].

2.2.4 Washing hands, a preventive measure to control the spread

Center for Disease Control and Prevention recommends hand cleansing regularly to prevent on-spread of COVID-19.

- Wash your hands- often with soap and water for at-least 20 seconds, especially after the person has been in a public place or after coughing/sneezing.
- If soap and water are not readily available, then the use of a hand sanitizer that contains at least 60% alcohol is recommended. All surfaces of the hand should be covered and rubbed together until they are dry.
- Avoid touching the eyes, nose, and mouth with unwashed hands [33].

2.3 Hand hygiene compliance

There are five 'moments' for hand hygiene which are the indicators for hand hygiene compliance as per the World Health Organization [34]. As shown in Figure 2.3.1, they are:

1. Before touching a patient
2. Before clean/aseptic procedure
3. After body fluid exposure risk
4. After touching a patient
5. After touching patient surroundings

Auditing/compliance procedures check if these '5 moments of hand hygiene' are followed by health care workers. Direct observation is the golden standard to measure hand hygiene compliance, where an expert evaluates and approves the occurrence of hand washing activities for the given moment/ opportunity. Direct observation helps to assess the strength or weaknesses in hand hygiene behavior, identify the number of hand-hygiene opportunities, their indications, assess the technique for washing hands and provide feedback to healthcare workers.
Direct observation is enhanced with the use of mobile hand-held devices instead of traditional pen and paper methods to capture hand washing data [35].
Measurement of the soap usage, conducting a survey, video monitoring and electronic surveillance are additional methods being checked during the direct observation [36]. Video monitoring systems include installation of video cameras at the sinks, entrance of the surgical department in order to monitor hand hygiene compliance [35]. However, there are considerable constraints with implementing direct observation in practice, as it is time consuming and requires training to be given to the health care experts. There might also be personal preferences involved that may result in an unfair inspection [37].
The Hawthorne effect where the participant is aware of being observed and might alter their natural behaviour and interobserver variation are another set of limitations for direct observation [36]. Video monitoring systems with video cameras installed at the sink or at the entrance of the surgical department does not necessarily suffer from the Hawthorne effect but requires an expert for the evaluation [35]. Although direct observation is the golden standard for measuring hand hygiene compliance rates, there also exists technical solutions to assess hand hygiene compliance in the literature.

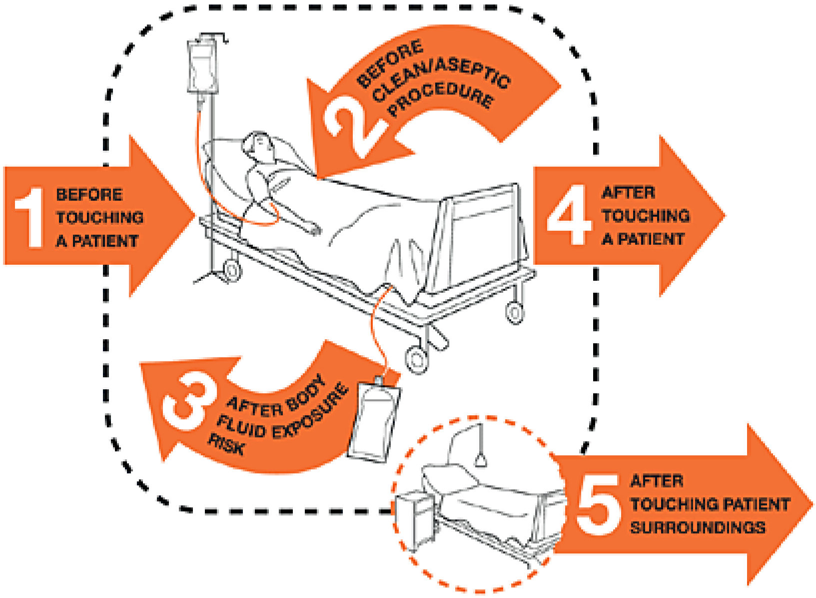

Figure 2.3.1: Hand hygiene moments-when should you wash your hands?[20]

2.3.1 Technical approach to assess hand hygiene compliance

There are digital/ electronic hand wash monitoring systems to measure hand hygiene compliance. These systems generally monitor hand washing activity in two ways:

- Product usage: This involves the use of electronic counters to determine the number of times a soap dispenser is pressed. Electronic counting devices are placed under the hand rub dispenser to measure the product usage and count the number of hand wash episodes that have occurred in a hospital [38, 39, 40]. A study has been conducted to determine the changes in hand hygiene frequency and found that the use of dispensers increased during the medical rounds, visiting hours and day shift versus night shift [35].
- Location based reminder systems: These systems alert the user to wash their hands when they are around susceptible locations such as patients' beds. These solutions monitor the number of missed hand hygiene events and the amount of time taken in washing hands. [35, 38, 39, 40].

Standalone hand hygiene monitoring systems such as electronic counters, reminders and video monitoring are insufficient to record hand hygiene compliance and therefore require wearable devices and sensors for evaluation [35].

Mondol et al. developed a smart watch-based hand washing monitoring and reminder system that use Bluetooth beacons to access the user's location and Bluetooth enabled liquid dispensers for measuring the product usage [41]. Time taken for each hand wash activity is taken

into account to ensure that the hands are washed for an optimum amount of time.
As mentioned, the WHO guidelines recommends minimum of 20-30 seconds of hand wash [20]. Philip et al. measured the product usage with the use of motes battery powered programmable devices. Motes or sensor nodes are worn by the health care workers, placed in a patient's rooms and attached to off-shelf dispensers. Time is recorded when the worker enters and leaves the patient's room [42].
These technical systems can be a substitute to or in addition to the 'direct observation' as they are less likely to suffer from Hawthorne effect where users may modify their behaviour while being watched. However, high installation costs and a lack of correlation between the compliance rate generated by the direct observation procedure and automated systems is a drawback with these existing technical solutions [40, 41, 42].
Another significant disadvantage of the above-mentioned technical solutions is that the structured WHO guidelines for washing hands are not assessed and therefore there is no knowledge about the quality of the hand-wash conducted by the health care workers.

2.3.2 Low compliance rates for hand washing

The literature emphasizes that the adherence to best hand hygiene practices is sub-optimal in health care environments worldwide [43, 44, 45, 46]. Continuous improvement strategies and intervention programs such as "clean your hands" initiatives are conducted to increase awareness and the compliance rate [44, 45, 47]. However, the process of evaluating the rate of compliance varies through geographical location. The commonly accepted method for measuring hand hygiene compliance is counting the number of hand hygiene events per hand hygiene opportunity [47]. It is noted that nurses have a higher compliance rate as compared to physicians and other health care workers [48, 49]. It is also observed that the health care workers are non-compliant towards a particular hand hygiene moment-"before getting in touch with the patient or/and the environment" [47, 50].
Pittet et al. studied the factors influencing noncompliance among health care workers [25]. They are:

1. Occupational Role: Being a physician or a nursing assistant rather than a nurse.
2. Gender and Occupation: Being a male nursing assistant in an Intensive Care Unit.
3. Workload pressures and under staffing.
4. Perception of the alternatives such as wearing gown and gloves; use of an automated

sink may interfere with normal working conditions and infrequent use due to the fear of contamination and confusion regarding the functionality of the system [46].

There is a possibility to increase the rate of hand hygiene compliance by introducing a digital application that can track user's hand washing gestures and provide feedback whether the user has performed the hand-washing activity to an appropriate level. Additional personalised game/ motivational design features may enhance the user engagement with the application. Positive reinforcement and feedback can be helpful in motivating users to wash their hands.
In the next section, various hand hygiene stages as per health authorities such as WHO are discussed in detail.

2.4 Guidelines for hand hygiene practices

There are 11 sequential steps involving two hand gestures in the 'hand washing' guidelines as per the World Health Organization(WHO) and shown in Figure 2.4.1 [20]. They are as follows:

Stage 0: Turning on the faucet and wet hands with water.
Stage 1: Press the dispenser and apply enough soap to cover all hand surfaces.
Stage 2: Rub palms of both hands in rotation.
Stage 3: Right Palm over the left dorsum with fingers wide open, and interlaced.
Stage 4: Palms of both hands in contact with fingers interlaced.
Stage 5: Fingers of both hands are interlocked.
Stage 6: Rotational rubbing of the left thumb clasped in the right palm.
Stage 7: Rotational rubbing backwards and forwards with the clasped fingers of right hand in the left palm and vice versa.
Stage 8: Rinse hands with water.
Stage 9: Dry hands thoroughly with a single use towel.
Stage 10: Use towel to turn off the faucet.
Stage 11: Hands are clean and safe.

Based on these hand hygiene stages, it is observed that:
Stages 2-7 directly involve hand washing with unique hand movements.
Other stages such as turning on the tap and getting soap are prerequisites in the hand washing process and are explored in detail in the next chapter.

Figure 2.4.1: Steps involved in hand hygiene practices [20]

Two-hands dynamic gestures in hand washing need to be further broken down into unique traceable features for the purpose of tracking hands.

To track dynamic hand washing movements, there is a need to explore the existing literature and the technical work that has already been done in the field of hand gesture recognition.

2.5 Methods of hand gesture recognition

Hand gesture recognition is an active research field which tries to capture and analyse gestural movements within the paradigms of human computer interaction.

The most common gesture acquisition methods are:

Vision based recognition: The gesture is captured with the help of a web camera or a depth camera.

Device based recognition: Use of wired/wireless gloves to detect hand movements of the user and motion sensing input devices to capture hand gestures and motions [51].

Given below are an example of some of the data acquisition tools in the field of hand gesture recognition.

1. 3D sensors for tracking hand movements:

   Microsoft kinect: It contains an RGB camera and a depth sensor using an Infra red light source that can produce three dimensional positional data in real time. Kinect data is acquired as a stream of two 640 * 480 images at 30 frames per second(fps). One of them is an ordinary 24 bit RGB video image and the other is an 11 bit depth image from which (x,y,z) coordinates for a hand movement can be calculated [52].

   Leap Motion Controller: It is a low cost, lightweight (45 grams) USB peripheral device with two inbuilt monochromatic IR cameras and three infrared light emitting diodes (LEDs), designed for hand and arm tracking at an approximately 60cm (viewing range). It gives 3D positional data for hands and arms in real time at 60fps [53, 2].

2. RGB-images/videos: With the help of image processing techniques, hands can be extracted from the video recordings and hand-motion can be detected for the purpose of gesture recognition [54].

3. Wearable devices: Hand gloves or data gloves can be worn on the hands which facilitate sensing and motion control [55]. RFID tags can be attached to the user's clothing. The tag reading in an RFID environment can be used to recognize the user gestures and to enable intuitive human computer interaction [56].

Data acquisition is the first step in building a robust gesture recognition model.

Hand gestures have a wide number of applications in the context of human computer interaction where they provide nonphysical contact and control complex virtual environments with a more intuitive user approach.

2.5.1 Hand gestures and associated features

Hand gestures are a natural mode of communication and used in diverse areas such as sign language, aviation industry, music directions and virtual games.
Hand Gestures can be classified as:-

- Static Hand Poses: A hand posture at some time instance.
- Dynamic Hand Movements: A sequence of postures connected by motions over a short time span [55].

Hand gestures can be further broken down to extract the hand features, such as number of fingers, thumb status, skin color, alignment of fingers, and palm position. For the purpose of tracking a gesture, a feature is defined as characteristic information of the local appearance of the object to be tracked [57].
Geometric features include the form and shape of the hand; such as fingertip distance, elevation, curvature, palm features and hand centre [58]. Non Geometric features may include skin colour and motion [58].
Features can also be classified as:

- Low-level features: include elements such as lines, corners, and edges of a hand. They correspond to the minor details of the object to be tracked. These features are useful for building an overall hand detection model.
- High-level features: require spatial information of the object such as fingertip position; can be calculated or composed from low-level features. These include finger location,centre of the palm. These features can be useful in the classification of dynamic hand movements.

Combinations of low-level and high-level features are useful in developing the overall gesture recognition system. Low-level features can be extracted from 2D images and high-level features can be utilized from low cost, low-power consumption - 3D sensors such as the Leap Motion Controller, and Microsoft Kinect [15].
Figure 2.5.1 demonstrates the sketch of hand features for the purpose of hand tracking and recognition and Figure 2.5.2 demonstrates the sequence of steps in building a successful gesture recognition system.

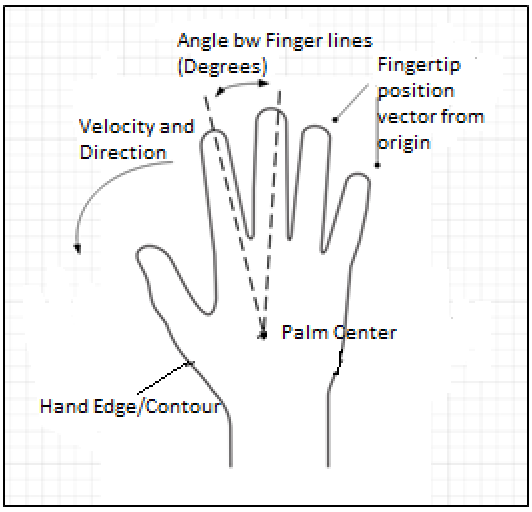

Figure 2.5.1: Low level hand feature such as edge detection and high level hand features

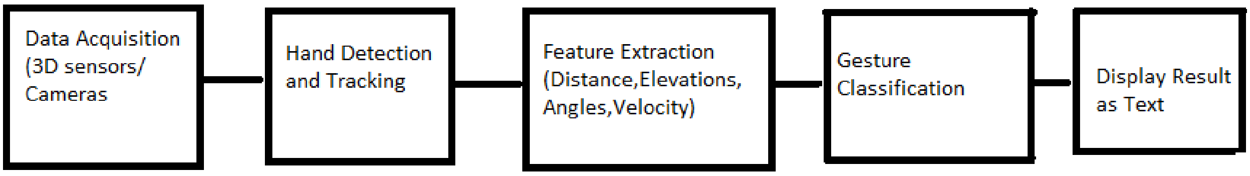

Figure 2.5.2: Hand gesture recognition model as proposed by Hamda et al. (2017) [58]

2.5.2 Gesture recognition applications with 3D sensors

Researchers have explored the suitability of gesture tracking devices for identifying various hand poses and movements.

- Fabio et al. extracted the hand region from the depth map and segmented it into palm and finger samples. Distance features between the palm centre and the fingertips were calculated to recognize various counting gestures [59].

- Lin Shao extracted the fingertip position and palm centre with the help of the Leap Motion Controller for tracking hand gestures. Distance between fingertip and palm centre and the distance between two fingers adjacent to each other was calculated. Velocity was detected to differentiate between static and dynamic gestures [60].

- Marin et al. used the Leap Motion Controller and the Microsoft Kinect jointly to extract the hand features such as fingertip position, hand centre, and hand curvature for recognizing the American sign language gestures [15].

- Jin et al. used multiple Leap Motion sensors to develop a hand tracking system for object manipulation where the sum of distal interphalangeal, proximal interphalangeal and metacarpophalangeal angles were taken into account. Strength of grasping and pinching were the object manipulation tasks that were recorded [61].

- Liorca et al. classified hand hygiene poses using a soft kinect camera [62, 63].

The previous work done shows the significance of 3D sensors for identifying various hand poses and gestures.

2.5.3 Gesture recognition with cameras

In this work, an investigation for the suitability of 3D sensor and cameras for tracking hand hygiene stages has been conducted. It is vital to understand the research in the field of gesture recognition with the help of camera images.

- Khan used color segmentation and template matching to detect American sign language gestures [64].
- Jophin et al. have developed a real time finger tracking application by identifying the red colour caps on the fingers using colour segmentation technique in image processing [65].
- Azad et al. extracted the hand gesture by image segmentation and morphological operation for American Sign Language gestures. Cross –correlation coefficient was applied on the gesture to recognise it with overall 98.8 % accuracy [66].
- Chowdary et al. detected the number of circles to determine the finger count in real-time, where in the scanning algorithm is independent of the size and rotation of the hand [67].

The study of past research in tracking objects and motion can be useful for tracking hand washing movements. Further discussion on image processing techniques are discussed in later chapters as they arise.

2.6 Introduction to deep learning

Deep learning is a branch of machine learning which is based on an artificial neural network(ANN) having an ability to mimic the behaviour of a human brain. Artificial neurons are the fundamental component for building ANNs. ANN consists of multiple hidden layers with multiple hidden units(neurons) or nodes [68]. It is an emerging approach and has been widely applied in traditional artificial intelligence domains such as semantic parsing, transfer learning, computer vision, natural language processing and more [69]. Over the years, Deep learning has gained increasing attention due to the significant low cost of computing hardware and access to high processing power (eg-GPU units) [69]. Conventional machine learning techniques were limited in their ability to process data in its natural form. For decades, constructing a machine learning system required domain expertise and fine engineering skills to design a feature extractor that can transform the raw data (example: pixel values of an image) into a feature vector, which is passed to a classifier for pattern recognition [70]. Deep learning models learn features directly from the data without the need for building a feature extractor. "Deep" usually refers to the number of hidden layers in the neural network. Traditional neural networks contain 2-3 hidden layers where as deep networks can have as many as 150 hidden layers and so it requires high computing power [71].

While deep learning techniques are not used directly in this work but any gathered hand washing dataset could be further utilised in this field.

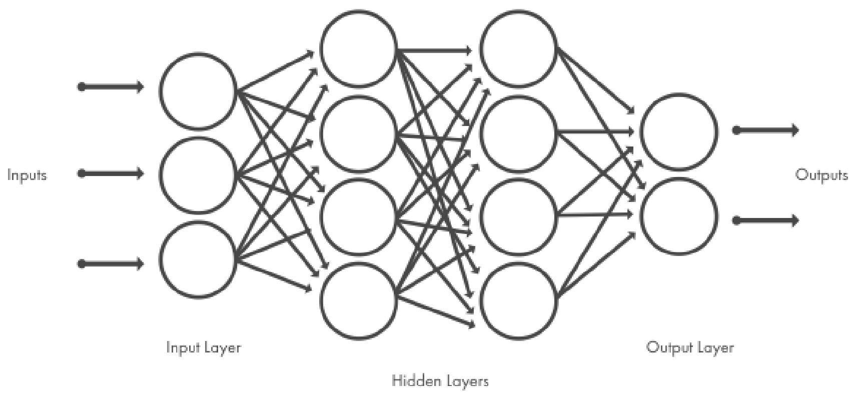

Figure 2.6.1: Neural networks, with interconnected nodes. Number of hidden layers may vary from tens to hundreds [71]

2.7 Successful applications based on deep learning

As deep learning continues to mature, there are many deep learning applications ranging from language recognition, self-driving cars to medical research classification of CT images for lung nodules [72]. Few applications in the field of Computer vision and pattern recognition that gained popularity with the use of deep learning models are:

1. Automatically colorize gray-scale images: Convolution neural network is utilized to jointly extract local and global features from an image and then fuse them together to perform the final colorization of black and white images [73].

2. Generating automatic natural language description of images and their regions : Deep neural network model is developed that associates segments of text with the region of the image that they describe in order to create a label for the images [74].

3. Text recognition : Recognizing and Retrieving text in natural scene images- text based image retrieval [75].

4. Visually indicated sounds: Recurrent neural network is developed to predict sound features from silent videos by analyzing the visual scene [76].

5. Deep photo style transfer: Photographic style transfer seeks to transfer the style of a reference style photo onto another image. The deep network is built to transfer colors from the input image and avoids spatial distortions [77].

2.8 Convolution neural network CNN

CNN is a type of deep learning model for processing data that has a grid pattern such as images and primarily used in the field of pattern recognition within images. CNN consists of three types of layers. They are Convolution layers, pooling layers and fully connected layers. Convolution and pooling layers perform feature extraction while a fully connected layer maps the extracted features into the final output [78, 72]. A CNN architecture is formed when these layers are stacked together.

- Convolution layers: It is responsible to extract low level features such as edges, color from the input image with the use of a convolution filter which traverses through the entire image. With added layers, the architecture adapts to high level features building a network that understands the overall images in the data set.

- Pooling layers: It is responsible for reducing the spatial size of the given input thereby reducing the number of subsequent learning parameters; flatten the image and transform into a 1D array of numbers (or vectors).
- Fully-connected layers: It is responsible for connecting one or more fully connected layers, also known as dense layers, in which every input is connected to every output by a learnable weight [72].

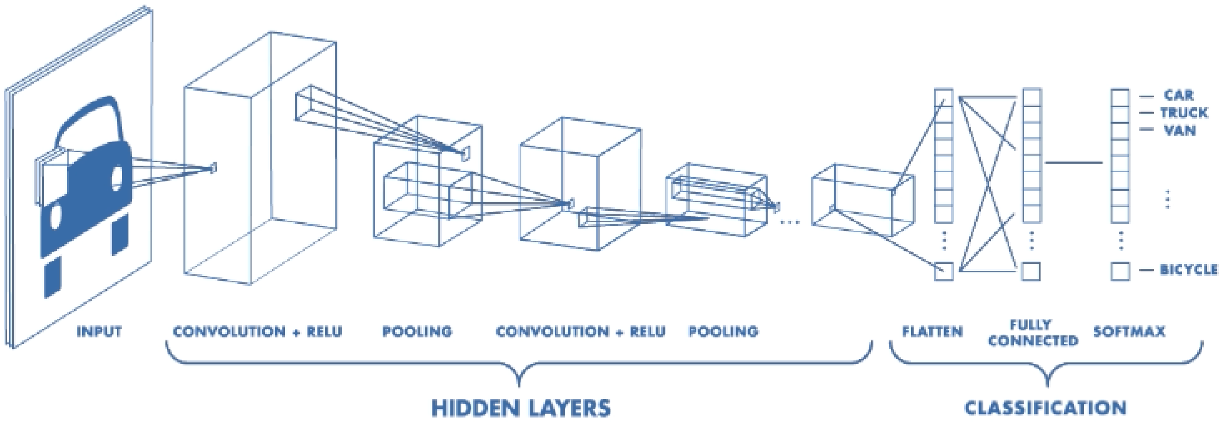

Figure 2.8.1: CNN-architecture [79]

2.9 Transfer learning

The future work will focus upon the classification of various hand movements with the help of the video recordings of users washing their hands. As a pilot study, single hand gestures will be utilised for the classification at first. The main challenge of this work is the acquisition of "large-scale data sets". Deep learning models essentially require thousands of data samples and heavy computational resources such as GPU for accurate classification and prediction analysis. However, there is a branch of machine learning, popularly known as "transfer learning" that does not necessarily requires large amounts of data for evaluation.

Transfer learning is a machine learning technique wherein a model which is developed for one task is reused for the second related task. It refers to the situation where "finding" of one setting is exploited to improve the optimisation in another setting [80].

Transfer Learning is usually applied to the new data set which is generally smaller than the original data set used to train the pre-trained model.

Hussain et al. applied transfer learning to train caltech face data set with 450 face images with pre-trained model on ImageNet data set. Increasing the number of training steps (epochs) increased the classification accuracy but increased the training time as well. Computational power and time were the main limitations within the study [80].
Transfer Learning is an important tool in machine learning that can solve the basic problem of an insufficient training data. It tries to transfer the knowledge from the source domain to the target domain where training and test data is not required to be independent and identically distributed [81].
Yu-Chuan Su et al. applied transfer learning from image to video where the image corpus is weak labeled available in various social media such as Flicker, Instagram, and videos were manually labelled youtube videos [82].
The future work will explore the paradigms of transfer learning; pre-trained on image data set for the classification of hand hygiene video recordings which were created in this work.

2.10 Summary of the literature review

The overall literature review can be summarised as:

- Appropriate hand hygiene practices can reduce hospital acquired infections and so monitoring and detection of hand washing movements can prove to be beneficial in the health-care environment.
- Low hand hygiene compliance rates in health-care settings around the world shows that poor/no hand washing is a global problem.
- There are technical solutions that involve electronic counters and reminder systems for monitoring if hand washing practice has occurred. However, they do not assess if all hand hygiene stages were performed. The quality of the hand wash is not taken into consideration as well.
- Tracking hands and data acquisition for recognising hand gestures take place via 3D gesture trackers, RGB cameras, and wearable gloves.
- Hand gestures can be classified as a static hand pose (image) or a dynamic hand movement that involves motion-speed and direction.
- Hand gestures can be further broken down into features. Low-level features are generic and correspond to lines, corners, and edges of a hand. High-level features are fine details

of a hand such as fingertip location, center of the palm and palm orientation.

- Researchers have used 3D gesture trackers in other applications such as recognising sign language gestures [15] and for recording the object manipulation tasks such as strength of grasping and pinching [61]. This shows the significance of 3D gesture trackers in building a hand gesture recognition system.

- Researchers have also explored the field of computer vision for gesture recognition in the past. For examples, hand gestures were extracted from the American sign language gestures by Azad et al. (2013) research [66].

In the next chapter, hand washing movements were critically analysed and hand features associated with each movement were extracted and recorded. In addition, an aluminium rig was constructed to fit it to the laboratory's sink and imitate the real hand washing setup.
A video data set is created by capturing the hand washing movements of participants. It is also included in the next chapter.

Chapter 3

Video Analysis of Hand Hygiene Experts and Feature Identification

In this chapter, the process of hand washing has been carefully analysed by examining the video recordings of healthcare workers washing hands according to structured guidelines. The hand features associated with every hand washing step were observed and recorded. Figure 3.0.1 illustrates the methodology used and basic work flow diagram where the hand features were extracted from video analysis. These features will later be required to be detected by 3D sensor and cameras in later chapters.

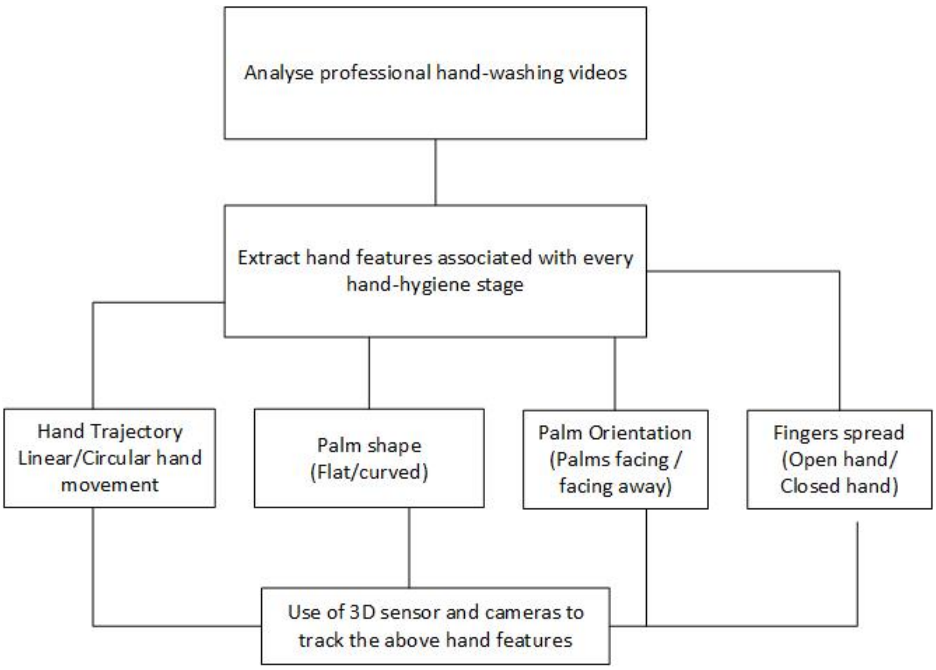

Figure 3.0.1: Elementary work flow diagram

3.1 Collection and observation of hand hygiene videos

There are structured hand washing guidelines with a sequence of steps to be followed by healthcare workers. It was important to observe and understand every hand movement involved in hand hygiene activity in order to automatically detect the hand hygiene stages with the goal of providing feedback. It was necessary to analyse hand movements in such a way that we can extract useful information in order to distinguish one movement from another. The hand movements may be further broken down into unique identifiable features that can be fed to a classifier for the classification of various hand hygiene movements as future work.

Videos selected: 10 video recordings of professional healthcare workers were selected for analysing hand hygiene stages. These videos were published by hospitals and health centers. It is important to note that the videos were sourced from reputable authorities where healthcare workers are trained to wash their hands according to the WHO guidelines. This kind of video-based observation has been adopted by many researchers previously. It is a common method used by the primary care researchers to analyse complex doctor-patient interactions and act as a source for a richer data collection process [83]. The idea behind the video based observation is that if the variations can be observed by a human observer then technical solutions can be developed to identify those variations and further classify hand hygiene movements.

The sources for the videos were recognised healthcare centers and hospitals across the world. Hand hygiene guidelines are published from the World Health Organization and must be followed by healthcare workers worldwide and so the videos were selected globally. The sources were

- WHO official hand hygiene video
- NUH, infection control, UK
- NHS barnsly, UK
- Eddy visiting nurse association, USA
- Cypress health, Canada
- University of Leicester, UK
- Vygon Ltd -leading supplier of surgical and medical devices, UK
- Symmetry hand hygiene Ltd, US-UK
- Online nursing forum

- Saskatoon health care, Canada

Each video comprises 11 sequential hand washing steps and the video length was in the range of 30-40 seconds. Some healthcare workers have performed additional steps such as nail cleaning and wrist washing. Every hand hygiene movement was carefully observed in each video recording and the unique hand features associated with the hand movements were extracted. The video recordings were segmented to observe each hand washing stage carefully, to note the pattern followed by the health care workers and their interpretation of the hand washing guidelines.

3.2 Distinct features extracted from the video

Five key features were identified through the video-based analysis. These key features were noted for every hand hygiene stage in all of the ten videos. The hand hygiene stages were discussed in detail in the previous chapter.
The features are as follows:

1. **Palm orientation:** Palms of both hands are either facing each other (stage 7,5,4,2) or facing downwards, also referred as pronation (stage 6,3) or facing sideways (stage 6).

2. **Palm shape:** Palms of both hands were found to be flat or curved.
   Both palms are flat: Stage 4,3,2
   Both palms are curved: Stage 5
   One palm is flat and one palm is curved: Stage 7,6

3. **Finger spread:** Various combinations of 4 entities were observed for fingers in all hand hygiene stages.
   Straight: When fingers are flat.
   Curved: When fingers are bent.
   Open: When fingers are at a wide distance from each other.
   Closed: When fingers are in contact with each other.

4. **Hand Trajectory:** The hand movement in all the video recordings for all HH stages was observed as a Linear motion (one- dimension along a straight line); (stage 4,3,2) or a non-uniform circular motion (stage 7,6)

5. **Frequency and time:** The frequency in Hz was computed as

$$f = numberofcycles/timeduration \tag{3.1}$$

The recommended time for the process of hand washing is 20-30 seconds [20] but there were variations with longer video-lengths. However, looking at the frequency for each stage, it is found that the workers evenly distributed the overall hand washing time among all six hand hygiene stages.

Table 3.1 lists the hand features associated with each Hand Hygiene stage through video observation.

Given below is the summary of the useful findings from the hand hygiene video analysis. These findings will further be explored in the next chapter to test the suitability of a hand gesture tracking device, Leap Motion Controller for detecting these features.

1. **Hand trajectory** in all hand hygiene stages is linear hand movement or circular hand movement
2. **Palm shape of the hands** are either flat or curved
3. **Palm orientation of the hands** are in pronation, facing each other or facing sideways

Stage	Feature	Value
Palm to Palm(2)	Palm orientation	facing(10 videos)
	Palm shape	flat(10 videos)
	Finger-spread	straight,closed(10 videos)
	Hand trajectory	linear(6);circular(4)
	Frequency-time	0-3 Hz; 2-7 s
Right palm over left dorsum(3)	Palm orientation	pronation(8 videos)
	palm-shape	flat(8 videos)
	Finger-spread	straight,open(6 videos)
	Hand trajectory	linear(8 videos)
	Frequency-time	1-3 Hz; 1-10 s
Palm to palm with fingers interlaced(4)	Palm orientation	facing each other(8 videos)
	Palm-shape	flat(8 videos)
	Finger-spread	straight,open(4); curved,open(4)
	Hand trajectory	linear(7 videos)
	Frequency-time	0.8-2.5 Hz; 2-7 s
Opposing palms fingers interlocked(5)	Palm orientation	facing each other(7 videos)
	Palm-shape	curved(7)
	Finger-spread	curved,closed(7)
	Hand trajectory	circular(7)
	Frequency-time	1-3 Hz; 2-10 s
Rotational thumb rubbing(6)	Palm orientation	pronation(7 videos)
	Palm-shape	flat(5 videos)
	Finger-spread	curved,closed(4)
	Hand trajectory	circular(7 videos)
	Frequency-time	1-2.5 Hz; 2-10 s
Rotational rubbing with clasped fingers(7)	Palm orientation	facing each other(8 videos)
	Palm-shape	flat(8 videos)
	Finger-spread	straight,closed(4)
	Hand trajectory	circular(7)
	Frequency-time	0.8-3 Hz; 2-14 s

Table 3.1: In detail analysis of hand hygiene stages

3.3 Rig construction

A fixture rig was built, as shown in Figure 3.3.1, with the purpose of mounting cameras and the Leap Motion Controller near a sink. It was used for recording different hand movements, hand washing stages and in gathering videos for a later dataset.

Prototype: An Aluminium rig

Design: An aluminium rig was built with the dimensions of 1x0.8x0.8 m (LxWxH). The rig was tested to see if it can accommodate the Leap Motion Controller and a camera for gathering the data for this project.

The size of the rig was determined by various factors

1. The viewing range for the Leap sensor (40 cm-wide) and cameras
2. Controlled background exposure to avoid the skin coloured objects which can be misclassified as an actual skin. Green and white sheets may be used to minimise the unnecessary background information.
3. Maintain the anonymity of the user by focusing only on the hand movements. The height of the frame was reduced from 1 m to 0.8 m in order to avoid the appearance of body organs in the frame other than the hands.
4. Enough space to fit on a sink and accommodate utilities such as soap dispenser, hand-towel.
5. Lightweight yet sturdy so it can be relocated if necessary.

Experiments were conducted where an aluminium rig was used in tracking 'hand trajectories' with the help of the 3D gesture tracker, the Leap Motion Controller, and RGB-digital cameras. It was also used to detect the presence of a hand on the soap bottle and water tap in the hand washing environment by mounting the Leap Motion Controller on it. Figure 3.3.1 represents the experimental setup with the Leap and camera mounted over the frame. Figure 3.3.2 is the actual frame recording captured during the creation of the hand washing data set.

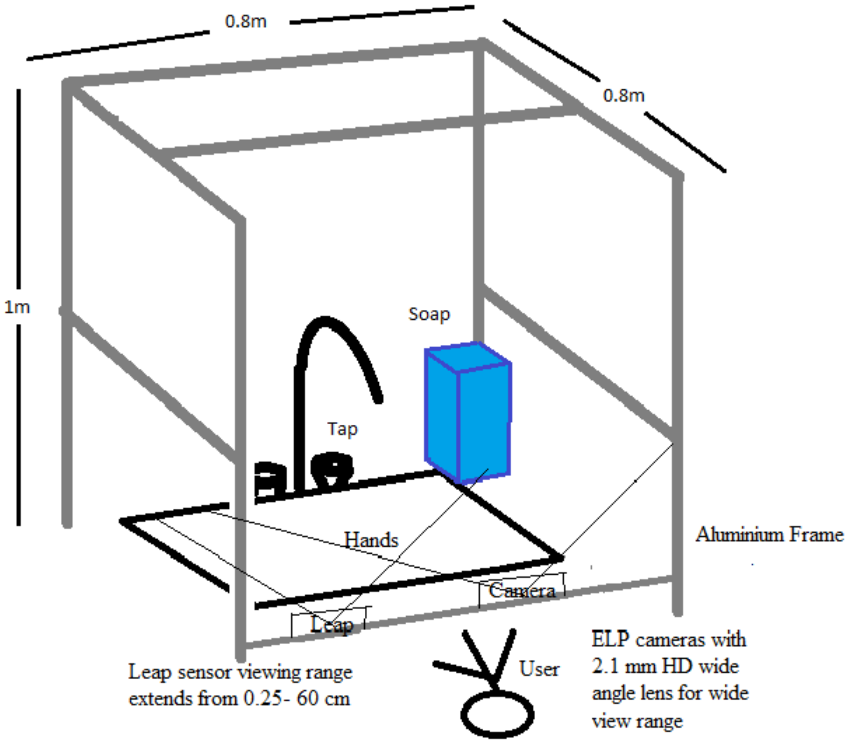

Figure 3.3.1: Experimental setup

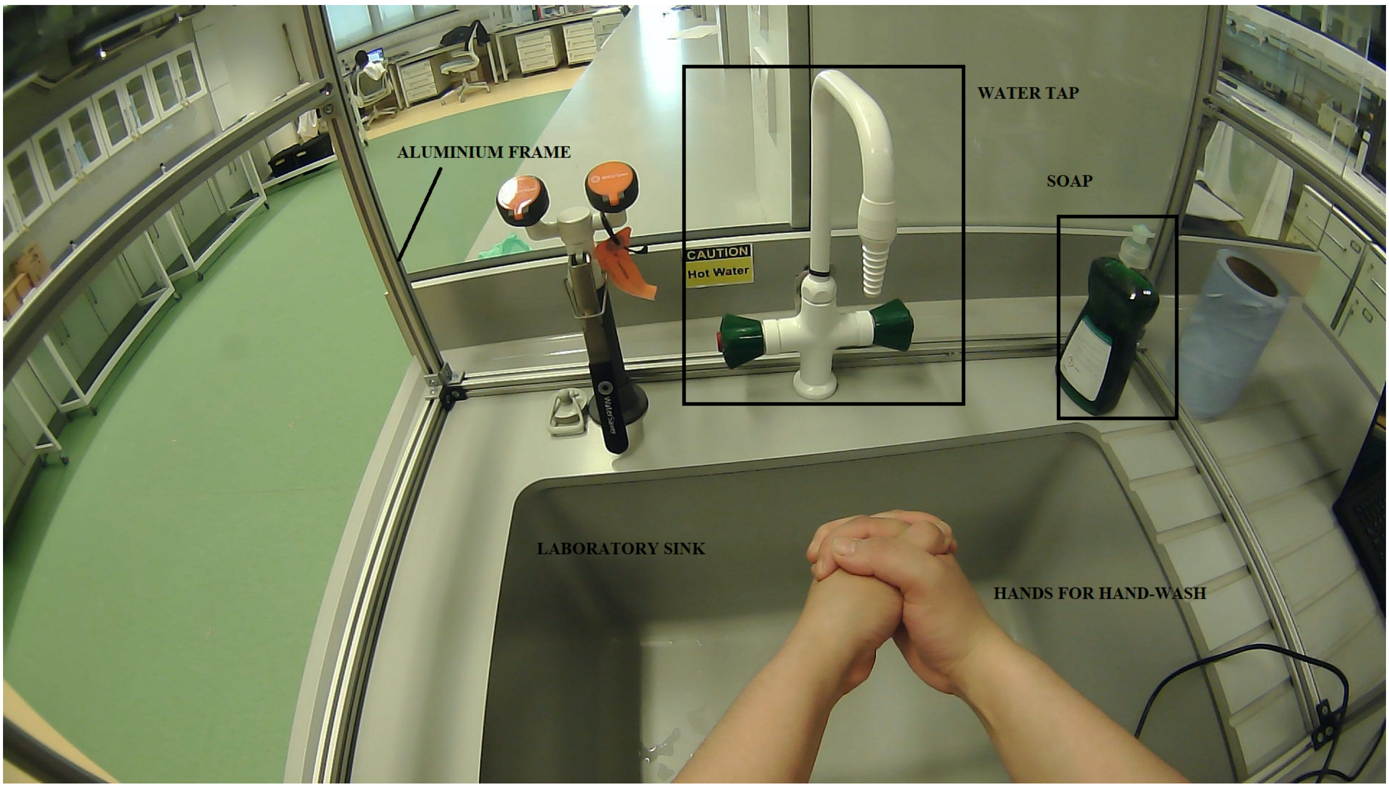

Figure 3.3.2: An aluminium rig is placed over a laboratory sink in the presence of soap and a water tap

3.4 Hand washing video repository

- The importance of having a video data set or repository was identified and 30 volunteers were asked to perform the various hand washing gestures in this project. In addition to the hand washing movements, one hand gestures such as linear and circular hand movements were also recorded.
- The aluminium rig was placed over the laboratory's sink and a digital camera was mounted over it. The camera was mounted in such a way that only the hand movements of the participants were recorded. This was done to maintain the anonymity of the user.
- Different ethnic background of the participants was helpful in creating a richer data set. It will be useful for testing the robustness of the system as a future work.
- The participants were informed about the study and were asked to complete an information sheet (attached in the appendix).
- Before recording the videos, the participants were informed about the importance of hand washing in a healthcare setup and WHO hand hygiene stages were demonstrated. They were able to refer to the hand hygiene poster while performing the hand washing movements. However, the participants were not enforced or trained so as to allow them to carry out the hand washing steps in a natural manner with their personalised interpretation of the guidelines. Use of running water was avoided in the data collection process.
- The video length for the hand washing activity was recorded for 25-30 seconds. Every hand washing step was followed by a pause where in the user was instructed to move their hands away from the camera.
- Video format for this data set is MP4 file with a size of range 40-60 MB and a frame rate of 29.84 frames/s.
- All of the six hand washing movements were recorded in one video for each participant.

An example of the collected information in addition to the video recording, stored in a csv file for all participants:
Gender = Male, Age = 29, Profession = Phd researcher, Country of origin = India, Skin tone = Brown, Video size = 62.9 MB, Video length = 29 seconds.
Video data collected were decomposed into individual frames/images and were utilised for tracking hand trajectories in chapter 5.

Chapter 4

Hand Washing Feature and Gesture Detection with Leap Motion Controller

4.1 Leap Motion Controller

The Leap Motion Controller is a low cost, lightweight USB peripheral device with two inbuilt monochromatic IR cameras and three infrared LEDs, designed for hand and arm tracking. The field of view is 150 degrees wide and 120 degrees deep, on an average 130 degrees, which is equivalent to roughly 60cm (viewing range) [53].
The data produced from the Leap sensor is in the format of (x,y,z) coordinates; directions and angles (pitch, yaw and roll with the Leap sensor as the center of reference. Placing a finger at the middle of the device will give the coordinates of a finger tip as [0,0,0].
The Leap sensor is able to track distinct hand and arm elements including bones and joints and all measurements are given in mm [53].
The Leap sensor can be used in different interactive applications such as touch less interfaces (kiosks), location based virtual reality experiences, healthcare (medical imaging) and robotics. It is easy to integrate into existing customer applications and it is certified compliant with safety and electrical regulatory standards.
Figure 4.1.1 demonstrates the Leap sensor internal coordinate system [53].

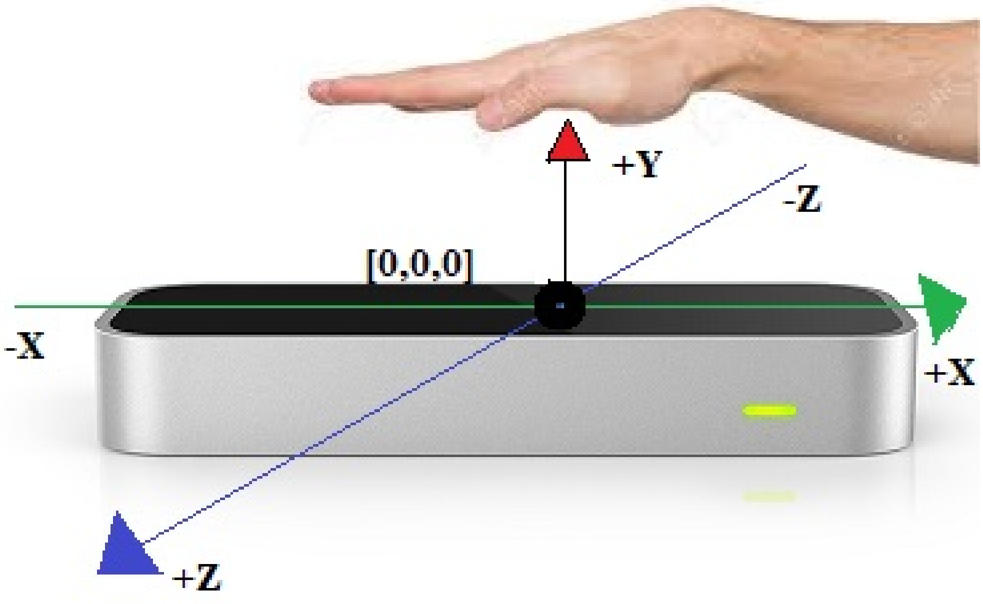

Figure 4.1.1: The Leap Motion Controller right hand coordinate system [53]

4.2 Suitability of the Leap sensor for tracking hand features

In this chapter, the Leap Motion Controller is utilised to detect various hand features as identified from the hand-hygiene video analysis. **The Leap Software Development Kit (SDK)** provides a range of inbuilt python functions that can be useful for tracking hands such as palm normal vector, palm position vector, hand grab strength (hand curvature). These functions can be used as the basis for tracking different hand poses and hand movements associated with hand washing. The Leap Motion Controller was placed horizontally at the bottom of the flat surface to test these features. The distance between the Leap Motion Controller and the hands was set according to the vendors guidelines, at 10 cm. Hand features observed through video analysis such as palm orientation, palm shape, hand trajectory and finger spread were detected with the help of the Leap sensor. These hand features were further coupled together to detect the basic hand hygiene stage - "rub hands palm to palm."

Platform: Leap SDK + Python Language

Python is an open-source object oriented language which is widely used for data analysis in current times. It is well supported with documentation and therefore chosen for this project.

Static hand poses: Static hand poses refer to gestures where hands are stationary. Experiments were conducted to extract the hand features such as palm orientation, palm shape, and finger spread with the help of the Leap sensor. The significance of these hand features in relation to hand washing is discussed in the previous chapter.

A) Palm Orientation- Palms of hands facing in the same direction/facing in the opposite direction:

The palm normal vector of the Leap sensor, is utilized to test if it can differentiate between hands facing each other or facing in the same direction. These are the two most common palm positions as discovered through hand hygiene video analysis. The Palm normal vector is a unit vector orthogonal to the centre of the palm. In an ideal scenario, the magnitude of the sum of the two opposite unit vectors is zero when the palms are directly facing each other. The magnitude of the sum of the two unit vectors is two when the palms are facing in the same direction. However, in real-time due to hand tremors and frequent loss of data frames, threshold values were selected. It means that the range was selected for the value of the sum as it was difficult to obtain an absolute 0 or 2 with two static hands recorded. The test is conducted to calculate the magnitude of the sum of palm normal vectors and detect if hands are facing each other or facing in the same direction. Equations 4.1 to 4.3 are applied for vector addition of left hand and right hand vectors with the usage of the Leap sensor. A vector in three dimensional space, A consists of x ,y, and z components (in mm). The resultant vector C is achieved by the addition of individual components of vector A to the components of vector B. An absolute value or the magnitude of a vector is represented for the experiments conducted with the Leap sensor.

$$\vec{A} = \vec{A}_x + \vec{A}_y + \vec{A}_z \tag{4.1}$$

$$\vec{B} = \vec{B}_x + \vec{B}_y + \vec{B}_z \tag{4.2}$$

$$\vec{C} = \vec{A} + \vec{B} \tag{4.3}$$

where $\vec{A}$ is the left hand palm normal vector and $\vec{B}$ is the right hand normal vector.

Result: The Leap sensor can clearly differentiate between the hands facing each other or facing in the same direction by the application of threshold values to the calculated sum of the vectors. Figure 4.2.2 is the Leap sensor data for 20 frames where the magnitude of the sum of palm normal vectors for both hands is represented.

The basic pseudo code applied for detecting the palm orientation

Read palm-normal vectors-left and right hand V1,V2

Calculate Sum=V1+V2

If absolute Sum<0.4

Then "Palm orientation: hands facing opposite to each other"

Else "Palm orientation: hands facing in the same direction"

Figure 4.2.1: An example of palms facing each other and facing in the same direction

The same hand pose is used in front of the Leap sensor for the detection.

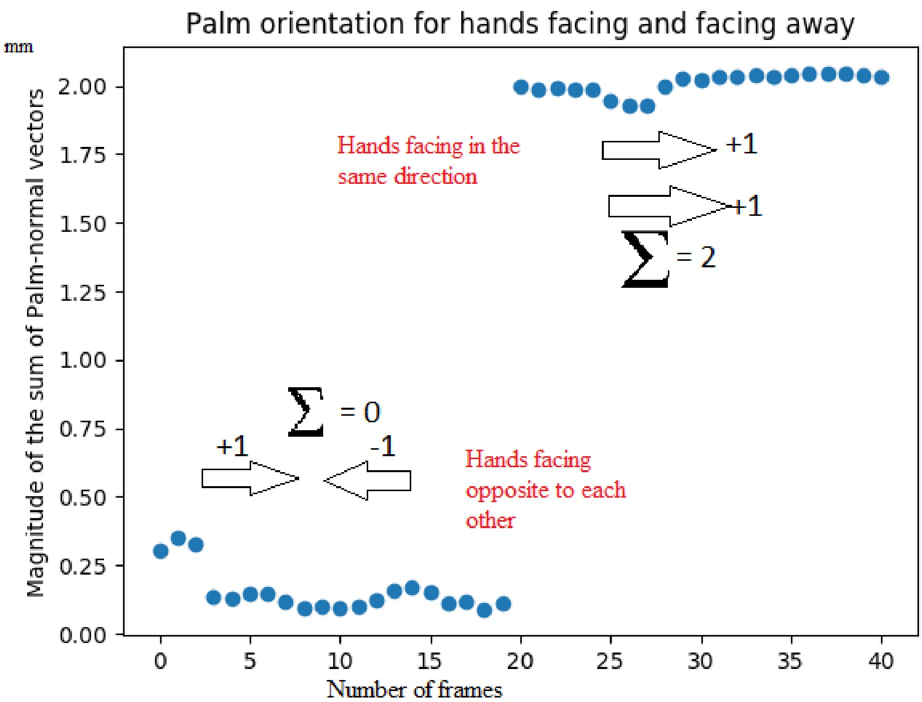

Figure 4.2.2: Palm orientation for the Leap sensor data set,

Data frames with the sum value of 0 represents 'hands facing each other' and the value of 2 represents 'hands facing in the same direction'

B) Palm shape-flat/curved: The finding from the video analysis shows that the palms of two hands are either flat or curved during hand wash and so the Leap sensor was tested to capture the hand curvature. The python based hand-grab strength function was utilized.

Result:

If hand-grab strength=0-0.3

then flat hand

else curved hand.

C) Fingers Spread:(in touch or not): This feature corresponds to the distance between two or more fingers. Fingertip position function in Python SDK was used to access the tip position value for an index finger and a middle finger. This feature determines if it is an open hand or a closed hand.

Result: The minimum distance between two finger tips was found to be 17mm beyond which the hand was considered as an open hand.

Criteria selected for an open hand:

Palm shape[flat]=0-0.3 AND the distance between fingers>17mm

Criteria selected for a closed hand:

Palm shape[curved]>0.3 AND fingers distance<17mm

Figure 4.2.3: The example of an open and closed hand

Dynamic hand movements: From the hand hygiene video analysis, it is noted that the most common hand trajectories in hand hygiene stage is linear hand movement and circular hand movement, and therefore the Leap sensor was tested to track the linear and circular hand trajectories in real-time.

Result: Linear hand movement The Leap sensor palm position vector gives the centre position of the palm (x, y, z coordinates) in millimetres. It was recorded with 1467 samples in 14 seconds with 104 frames per second (fps) for tracking one hand linear movement along the

X axis.

Circular hand movement 2101 samples were recorded with the Leap sensor in 18.9 seconds at 111 frames per second.

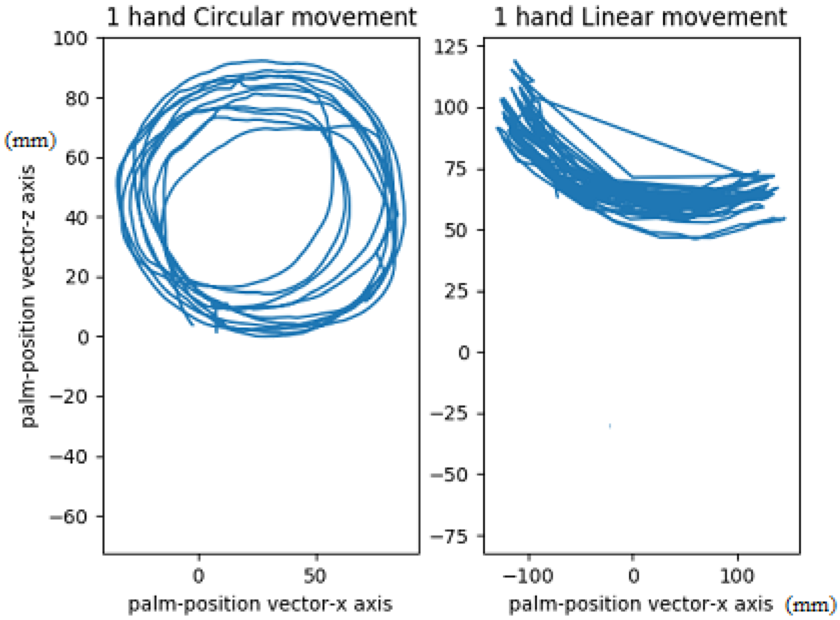

Figure 4.2.4: Plot for circular and linear hand movement for one hand data set.

Palm velocity: Python based three-dimensional palm velocity vector was used to determine if the hand is stationary or in motion. The magnitude of the palm velocity vector was calculated. In an equation 4.4, the 3 dimensional vector $\vec{w}$ is represented with the components x, y, z; where $\vec{w}$ is the right hand palm velocity vector. The formula for calculating the magnitude of $\vec{w}$ (in units), also known as the length of the vector is shown in equation 4.5. These formulae were utilised in real-time for recording one hand linear movement along the x axis of the Leap sensor coordinate system. Measurements of the palm velocity vector allow us to calculate the speed at which hands are performing hand washing actions. This is essential to note as the user might be imitating the hand washing guidelines but may not be washing the hands vigorously.

$$\vec{w} = (x, y, z) \tag{4.4}$$

$$|\vec{w}| = \sqrt{x^2 + y^2 + z^2} \tag{4.5}$$

where, $\vec{w}$ is the right hand palm velocity vector with x, y, z components.

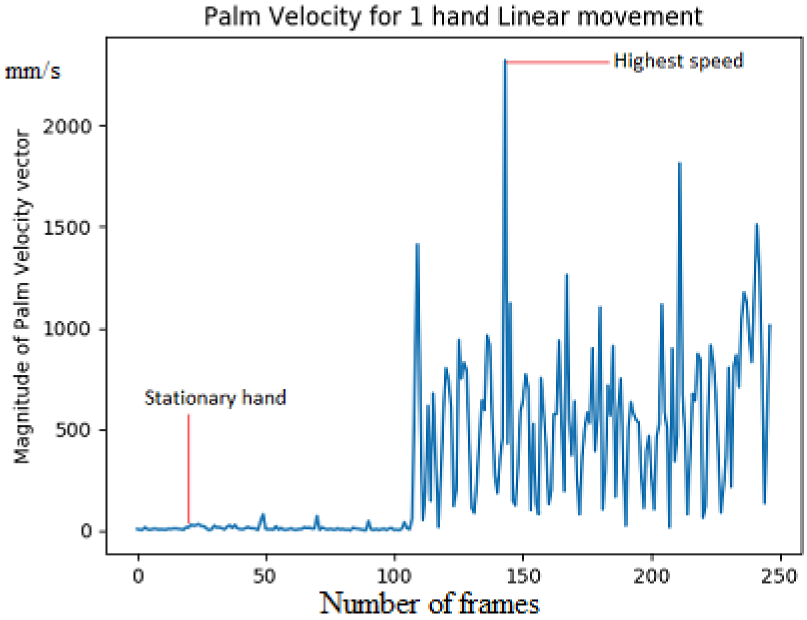

Figure 4.2.5: Magnitude of the palm velocity vector for one hand movement along X axis

The static hand poses such as flat hand or curved hand; palms facing in the same direction or palms facing in the opposite direction and dynamic hand movements such as linear and circular hand trajectories can be further integrated for the purpose of hand hygiene stage detection. In the next section, the most basic hand hygiene stage: "rub the hands palm to palm" is detected with the help of the Leap sensor.

4.3 Detect hand hygiene stage, rub hands palm to palm

The Leap sensor data for tracking hand features such as palm shape, palm orientation and finger spread were integrated. And the number of hands present in the scene was recorded to detect the execution and completion of WHO stage 2-rub hands palm to palm. The system tracks the presence of two hands, if they are facing each other. If not, it alerts the user. Then, it calculates the distance between the two hands and checks if it decreases gradually as hands comes in contact. In contact, the number of hands is reduced to one, as other hand is lost due to occlusion (discussed in detail later); it checks for the palm velocity for rotational movement and speed, then concludes the execution of hand hygiene stage2 - Rub hands palm to palm.

Euclidean distance between two palm-position vectors

Given two vectors:

$$\vec{u}, \vec{v} \in \mathbb{R}^n \tag{4.6}$$

The distance between two vectors or points in space is calculated as:

$$d(\vec{u}, \vec{v}) = \|\vec{u} - \vec{v}\| = \sqrt{(u_1 - v_1)^2 + (u_2 - v_2)^2 ... (u_n - v_n)^2} \tag{4.7}$$

where $\vec{u}$ is the left hand palm position vector and $\vec{v}$ is the right hand palm position vector.

Tests were conducted to find the maximum and minimum separation between two hands w.r.t. Leap sensor by calculating the euclidean distance between two palm position vectors.

Threshold values were selected to check if two hands are in contact with each other.

The sequence of steps involved in the detection of "rub hands palm to palm" stage are:

Read two hands in front of the sensor

Calculate the sum of palm normal vectors for left and right hand

If the magnitude of the sum is<0.4

. **Then** "Hands are facing each other"

. **Else** "Alert the user to align hands facing each other"

. **Calculate** the distance(D) between two palm position vectors for left and right hand

. **If** D<30mm

. **Then** "Hands are in contact with each other"

. **If** the magnitude of the palm velocity vector>150

. **Record** the duration of time>5 seconds

. **Then** Stage 2-"rub hands palm to palm" is detected

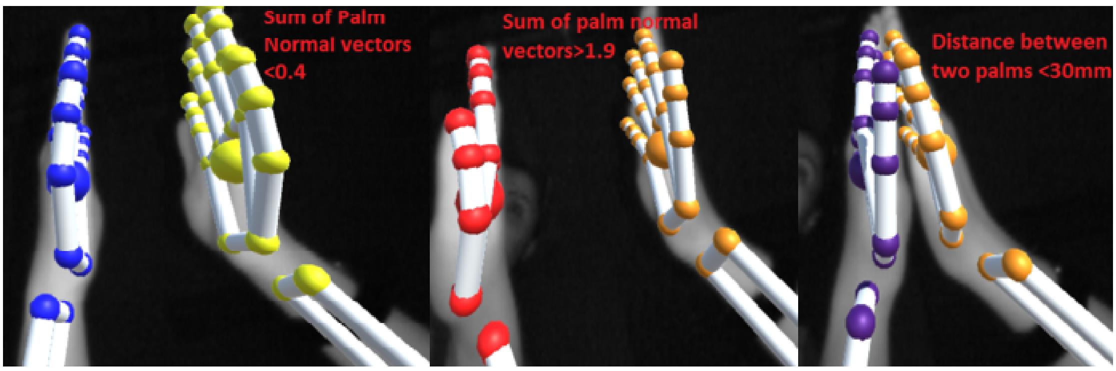

Figure 4.3.1: Hands facing AND not facing each other with threshold values. Hands are in contact with threshold values

4.4 Detect the presence of a hand near the soap dispenser and water tap

Soap and water usage are preliminary steps involved in the process of hand washing. All hand hygiene stages require the necessity of pressing the soap dispenser and turning on the water tap in a controlled environment. For this purpose- an aluminium frame was used to mount the Leap sensor and fit it onto the laboratory's sink within the reach of a soap dispenser and a water tap. Sequence of steps to detect if the user's hand has reached to the soap dispenser and the water tap are:

Read soap and water tap coordinates

Calculate an average of 10 frames for each

Calculate the distance (SD) between soap location and a palm position vector of a right hand

If SD <30 mm **Then** " User's hand has reached around the soap bottle "

Calculate distance (WD) between the tap location and a palm position vector of a right hand

If WD <30 mm **Then** " User's hand has reached at the water tap"

Soap dispenser and water tap location coordinates were preloaded in the script.

X,Y,Z coordinates of a soap dispenser w.r.t Leap sensor is (137.35,380.11,225.30)

X,Y,Z coordinates of a water tap w.r.t Leap sensor is (103.78,264.89,147.80)

Euclidean distance between the soap dispenser/water tap and the user's hand was calculated in real-time and when the hand approached towards the static locations of the soap/tap, the distance calculated was less than 30 mm.

A python code can be found in the appendix section of the document.

Figure 4.4.1: The presence of a hand on the soap is detected with the Leap sensor

The python script is added in the appendix section of the document.

4.4.1 Hand tracking under water

In reality, all hand washing stages are performed in the sink under running water, and therefore it is necessary to investigate if the Leap sensor can produce data frames under water. Optical sensors and infra-red light used in the Leap Motion Controller device does not restrain hand and arm tracking under the presence of water. This makes the Leap sensor an interesting tool for tracking hand movements in the context of hand washing.

- An aluminium rig was placed on the laboratory's sink and the Leap sensor was incorporated.
- The running tap water was used in conducting the experiment.
- The palm velocity vector from the Python SDK can be utilised to extract the data frames for stationary vs moving hands under the presence of water.

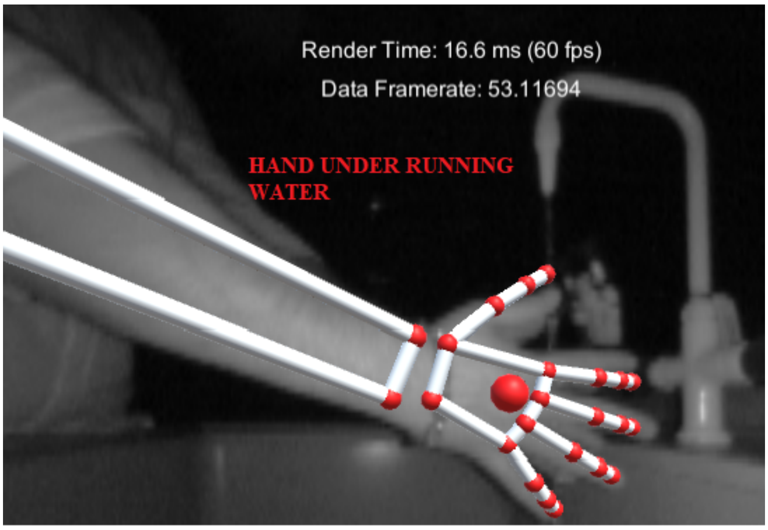

Figure 4.4.2: Tracking hand movement is achievable under the presence of running water with the help of the Leap Motion Controller

4.5 Conclusions about Leap Motion Controller

1. The Leap Motion Controller provides (x,y,z) coordinates in units of real world millimeters within the leap motion frame of reference. For example: if a finger's tip position is given as (x, y, z) = [100, 100, -100], then these numbers are in millimeters or x = +10cm, y = 10cm, z = -10cm.

2. The Leap sensor itself is the center of the frame of reference. The origin is located at the top, center of the hardware. If we touch the middle of the device, the coordinates of the finger tip will be [0,0,0].

3. The Leap Motion Controller is best suited for applications that involve the usage of one hand gestures such as sign language recognition, object manipulation tasks such as grasping and pinching.

4. The presence of water does not interfere with the working of the Leap Motion Controller and it can track hand and arm movements under the running water as shown in Figure 4.4.2. It was an important factor to influence the suitability of the device for tracking hand movements in hand washing application.

5. Hand occlusion is the major drawback for using the Leap Motion Controller to track two hands involved in hand hygiene stages. The key challenge is to distinguish the left hand from the right hand after the sensor recovers from the hand occlusion. Hands were occluded due to

    - Hands overlapping each other as in the case of HH-3, where one palm resides over the back of another palm. The Leap Motion Controller works best when both hands are in the direct line of sight. The hand is lost in scenarios where it is overshadowed by the other hand.

    - Hands are in contact with each other. One hand is completely lost when the hands are joined together.

Figure 4.5.1 demonstrates the "hand occlusion" when two hands are in the frame.

Table 4.1 briefly illustrates the hand features that can be detected by the Leap sensor.

Hand features	Detect 1 hand/2 hands in separation	2 hands in contact
Palm orientation	Yes(palm normal vector)	No
Palm shape	Yes(hand curvature function)	No
Finger spread	Yes(fingers tip position)	No
Hand trajectory	Yes (palm-position vector	No
Palm velocity	Yes (palm-velocity vector)	No

Table 4.1: Evaluation of the Leap sensor w.r.t. hand features

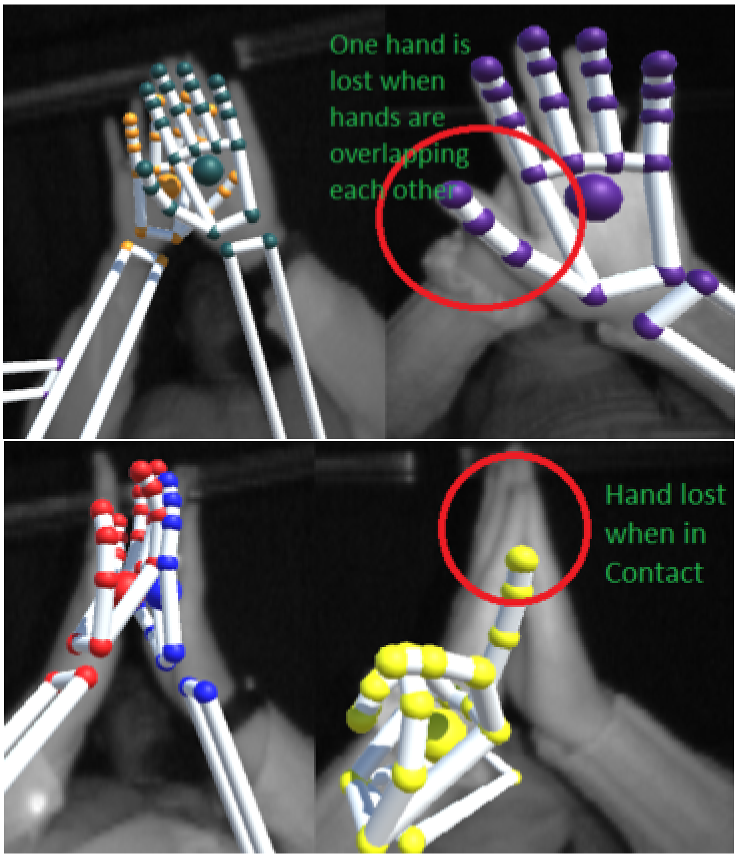

Figure 4.5.1: Hand occlusion: hand disappears when hands are overlapping or hands are in contact

Due to hand occlusion with the workings of the Leap sensor , RGB digital cameras were selected for further processing. The Leap sensor is suitable for extracting hand features with respect to one hand applications or two hands in separation. No support for contact tracking makes it a difficult choice for detecting hand features for hand washing. In the next chapter, image processing techniques are applied for the detection of two hands in hand hygiene stages.

Chapter 5

Hand Washing Feature and Gesture Detection with Digital Camera

Hand hygiene stages consist of dynamic two hand movements. These hand movements are quite complex and there is a necessity to break it down into more quantifiable gestures. To meet the purpose, hand washing video recordings were carefully analysed and the unique features associated with each hand hygiene stage were extracted.

The Leap Motion Controller was successful in tracking the hand features; static hand poses and gestures that involve one hand movement such as linear and circular one-hand trajectory. It could also detect the presence of a hand on the soap and water tap in a hand washing environment. Turning on tap and soap dispenser are the elementary steps in the process of hand washing.

The Leap sensor could also record hand gestures where one hand was clearly separate from another such as ”two hands facing each other”. However, the Leap sensor data was unreliable due to 'hand occlusion' when hands came in contact with each other. For example, ”rub hands palm to palm” hand movement requires two palms to be in touch. The Leap Motion Controller is limited in contact tracking and lost the wire-frame output when hands were in contact with each other. As hand hygiene guidelines involve both hands, it was important to explore the suitability of another kind of technology, other than 3D sensors. Digital cameras were selected for further investigation and tracking of hands within the context of hand washing.

Figure 5.0.1 lists the sequence of tasks accomplished in this chapter.

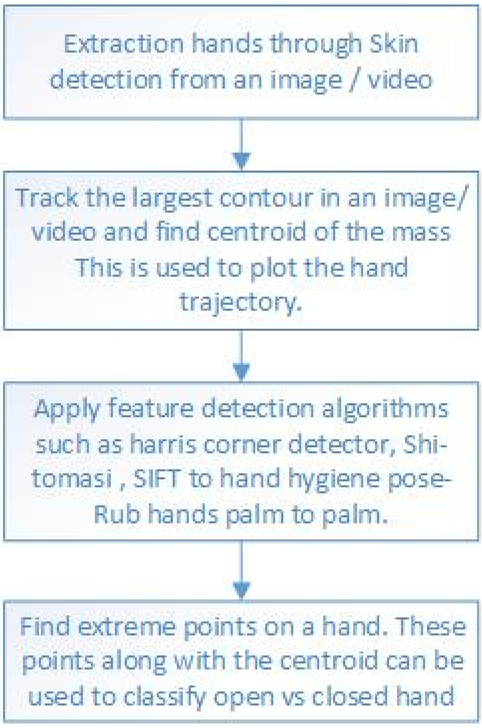

Figure 5.0.1: Sequence of tasks accomplished in this chapter.

5.1 Skin color detection in images

Skin detection is a process of finding skin coloured pixels in an image or a video. It is a pre-processing step to find regions with human faces and limbs in the images. There are various applications where skin detection has been applied in the past. For example, in one of the earlier applications, skin coloured regions were detected to identify nude pictures on the internet for content filtering [84].

In another application, skin pixels were detected to identify the TV news anchor for video annotation and retrieval [84]. In such an application, video frames largely contained the face and the hands of the anchor and the indoor background environment is controlled with hardly any skin tone colour related objects.

Skin detection becomes a challenging task in a background with skin tone colours such as wood, leather, skin coloured clothes, hair, sand etc. It becomes difficult for the model to differentiate between the actual skin and skin coloured objects and therefore the accuracy of the model may decline. Different lighting conditions can also affect the performance of the skin detection model due to its sensitivity towards illumination. Shadows in the images can cause poor performance as well. The accuracy of colour based skin detection models is also influenced by the camera characteristics. Skin colour in pictures taken by different cameras is not same even in the same lighting conditions [85].

Skin detection methods can be broadly classified into two types:- [86]

1. Pixel-based methods: where the colour feature is extracted from a pixel. Each pixel is classified as a skin or a non-skin.

2. Region-based methods: where the texture feature is extracted from a pixel and its surrounding neighbours. Spatial arrangement information of skin coloured pixels is detected. (Kruppa et al. ; Yang et al.) [87].

Pixel-based skin colour detection methods can be further divided into:-

1. Non parametric models: The main goal of non-parametric skin color models is to make an estimate about the skin colour distribution and allocate a probability value to distinguish between skin and non skin pixels in complex images. Histogram based look up tables, Bayes classifier and self organizing maps are known non-parametric methods [85, 88]. Non–parametric methods are fast in training and classification, independent to the distribution shape but requires high storage space and a representative training data set [89].

2. Parametric models: Requirement of a large training data set and high storage space leads to the need for more compact skin detection models-the development of parametric models. Parametric methods for skin color taxonomy include Gaussian model (single gaussian, mixture of gaussians) and elliptical boundary model where skin colour cluster is approximately elliptic in shape [89].

Use of different colour spaces for Skin Detection:-

Colour space is a mathematical model to represent colour information as three or four different colour components. There are existing colour spaces available for skin detection.

1. RGB based colour spaces:-

   The RGB colour space is the most common colour space in digital images. It encodes colours as an additive combination of 3 primary colours: red (R), green (G) and Blue (B). Figure 5.1.1 represents RGB color model. RGB colour space is simple to use but it does not separate the illumination channel-luminance from chrominance, and R, G, B components are highly correlated. The luminance of a given RGB pixel is a linear combination of the R, G and B values. Therefore changing the luminance will affect the R, G, B components and so this colour space is not very effective in varying lighting conditions [84].

Yet, Rehg and Jones have used RGB colour space to study the separability of the color space and yields satisfying results [90]. Normalized RGB is a representation that can be easily obtained from the RGB values as shown in the equations 5.1 to 5.3.

$$r = R/(R + G + B) \tag{5.1}$$

$$g = G/(R + G + B) \tag{5.2}$$

$$b = B/(R + G + B) \tag{5.3}$$

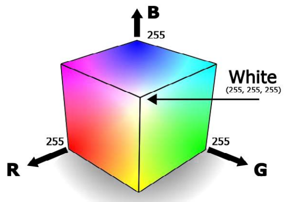

Figure 5.1.1: RGB Color Model [91]

2. Hue based color spaces(HSI,HSV,HSL):-

   These color spaces separates three components: the hue (H), the saturation(S) and the brightness (I, V or L). They can be mapped from RGB space via non-linear transformation. Figure 5.1.2 represents HSV color model. In case of HSV, as Hue varies from 0 to 1.0, the corresponding colours vary from red, through yellow, green, cyan, blue, magenta and 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 colours become increasingly brighter.

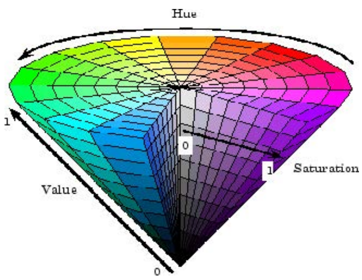

Figure 5.1.2: HSV Color Model [91]

3. Luminance based color spaces(YCBCr,YIQ,YUV):-

   These orthogonal color spaces are used in TV transmission. YCbCr is used in JPEG image compression and MPEG video compression. An advantage of using these color spaces is that the most video media are already encoded using these color spaces. Transforming from RGB into these color spaces is a linear transformation. YCbCr values can be obtained from RGB color space according to the equations 5.4 to 5.6. Figure 5.1.3 represents YCbCr color model. These color spaces can separate the illumination channel (y) from two orthogonal chrominance channels (IQ,UV,CbCr). Therefore unlike RGB, the location of the skin color in the chrominance channels will not be affected by changing the intensity of the illumination. The simple transformation and invariant to illumination intensity make these color spaces most appropriate for skin detection applications [84].

$$Y = 0.299R + 0.287G + 0.11B \tag{5.4}$$

$$Cr = R - Y \tag{5.5}$$

$$Cb = B - Y \tag{5.6}$$

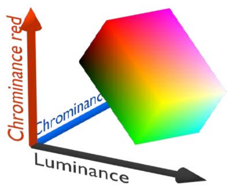

Figure 5.1.3: YCbCr Color Model [91]

5.1.1 Skin detection algorithm and results

A skin detection algorithm was applied to different kinds of hand images (single hand pose and two hand poses). For speed in development, hand images were preliminary captured with the help of a smartphone mobile camera and later with a ELP USB camera (2.0 Megapixel,30 fps). White and black background was selected for most of the images to prevent the mis classification of background pixels as foreground pixels.

YCbCr color model was selected for skin extraction. It is invariant to illumination intensity, yet consistent lights were chosen to prevent the capture of a noisy image (shadows and distortion). It is helpful in carrying out an accurate skin detection.

Skin detection algorithm was also applied to extract hands with different skin color tones to

ensure the robustness of the model with an intent that the algorithm should work for all kinds of users.
YCbCr color space model was selected and the threshold values were chosen based on the work of Jorge et al. [92].

Steps involved in skin detection algorithm based on YCbCr color space model are:

1. **Read** input frame in RGB format
2. **Convert** input frame to YCbCr format
3. **Apply** mask within the range,127>Cb>77 and 179>Cr>133
4. **Apply** gaussian blur to reduce the extra noise
5. **Save** the output frame with only skin pixels

Figure 5.1.4 represents the extracted skin for the hand hygiene pose 'palm to palm with fingers interlaced'.

Figure 5.1.4: Hands extracted for a hand hygiene pose based on YCbCr color model. White background was selected to avoid the mis classification of background as skin pixels.

5.2 Contours

A contour is a closed curve of points or line segments that represents the boundaries of an object in an image, having the same color and intensity [93]. Finding contours can be applied to determine the shape of an object (arc length; number of vertices), the number of objects in an image (number of contours) and measure the size of the objects [93].

Contour features

1. Image Moments: Image moments are statistical properties of a section of an image. Image moments can be used to extract useful information from the contour such as the centroid, area, etc. The moments are used as features for shape recognition.

2. Centroid: The centroid is defined as a coordinate (cx, cy) and is derived from the image moments.

3. Contour area: The contour area is the image area outlined by the contour.

4. Contour perimeter: Also called the ‘arc length’, this is the length of the contour in pixels.

5. Convex hull: The Convex hull of a shape or a group of points is a tight fitting convex boundary around the points or the shape of the object which is tracked. They are the minimum enclosing polygon of all points of the input shape. With a given set of X points in the Euclidean space, the convex hull is the smallest possible convex set containing these X points [94]. Convex hull of the simple polygon is shown in Figure 5.2.1

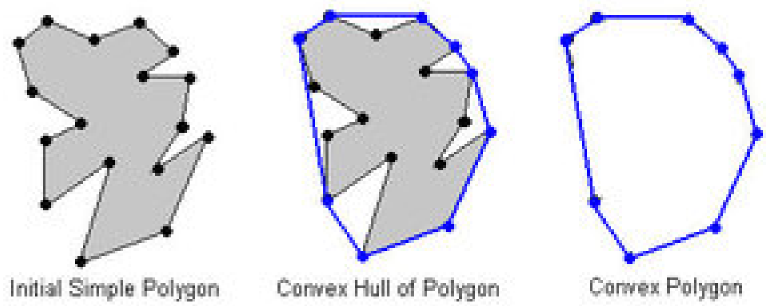

Figure 5.2.1: Convex hull of the simple Polygon [95]

5.2.1 Contour and centroid detection

Contour tracking is widely used in the field of computer vision/image processing. The idea behind contour tracking is to traverse the border of a region completely and detect the edge points. Xie et al. used the contour detection method to determine the number of copper cores in the wire [96]. Poda et al. computed the perimeter of a contour to detect a specific shape such as a pentagon, and the initial character ‘P’ is saved and transmitted to an Arduino for the movement of a mechanical arm [97]. Bochkarev et al. compares object characteristics such as area, perimeter and compactness of the contour of regular shapes to that of irregular shapes using Computer vision-open source library-OpenCV [98]. OpenCV offers cvFindContours function, which can retrieve contours from the binary image and return the number of detected contours. We have utilized OpenCV for detecting contours in video frames and finding the centroid of the largest contour.

5.2.2 Contour Centroid algorithm for tracking hands

Contour tracking broadly refers to traversing the border of a region completely and detecting the edge points. Contour tracking is useful for tracking an object of interest. In this work, the contours of the hands are detected and the subsequent centroid of the mass is detected. Further on, centroid plots were made for linear and circular hand movements.

Below are the given steps of the algorithm used to detect the contour and centroid for hand hygiene pose, rub hands palm to palm.

1. **Read** input frame in RGB format
2. **Convert** the frame to Gray-scale format.
3. **Apply** gaussian blur to reduce the extra noise
4. **Threshold** each frame
5. **Find** contours in the frame
6. **Sort** the contours and **find** the largest contour.
7. **Calculate** Image moments for the largest contour
8. **Find** x, y coordinates of the centroid; centre of mass of the hand
9. **Display** the frame with contour and centroid

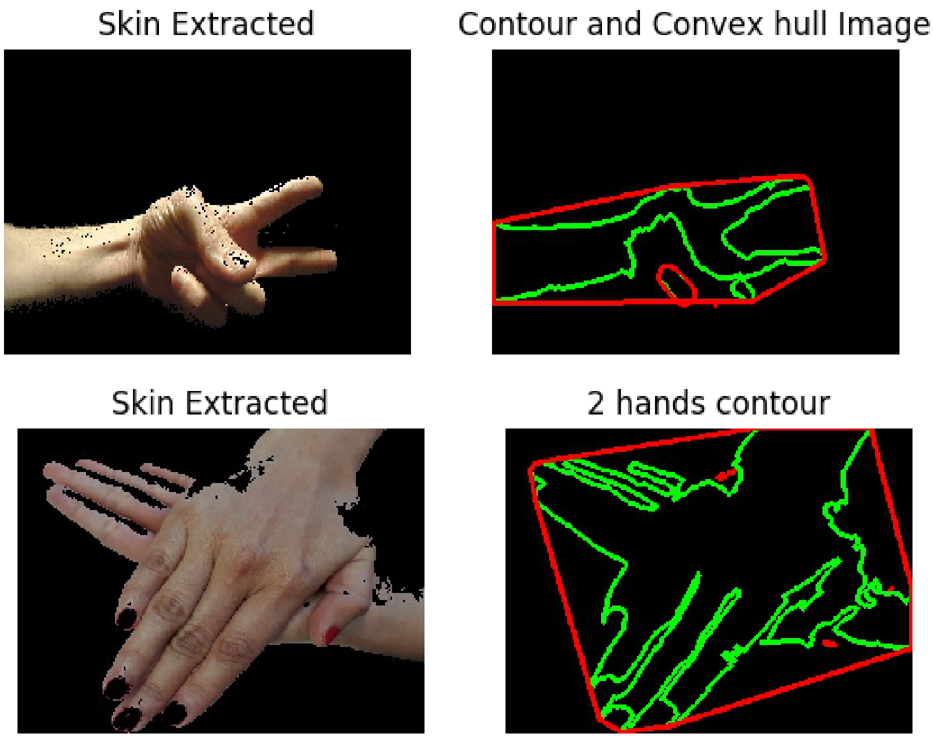

Figure 5.2.2: Contour detection for one hand and for the hand hygiene pose, rub hands palm to palm

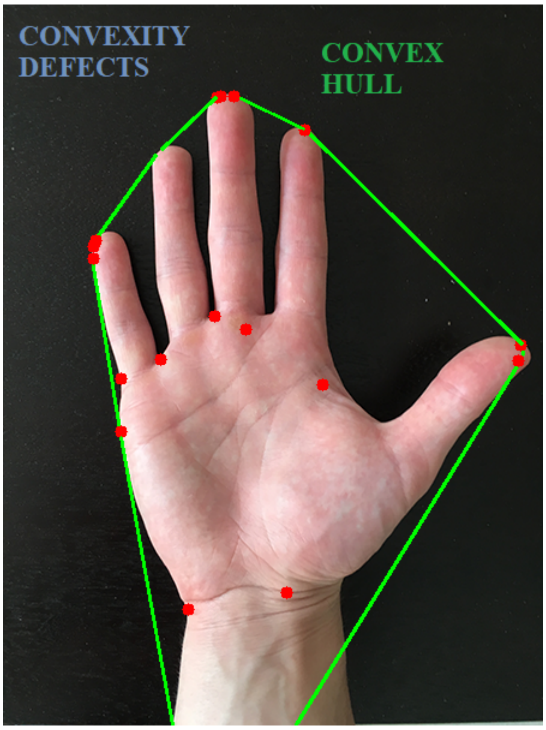

Figure 5.2.3: Convexity defect points and convex hull detected for an open hand

In Figure 5.2.2, the hand poses of users with different skin tones were captured. Images were taken against a neutral color background. Firstly, the skin extraction algorithm was applied to extract the hands, and then the contour centroid detection algorithm was applied to extract the overall contour/boundary of the hand.

Overall contour detection is useful for tracking hand movements. However, when two hands are in contact, contour detection results in a continuous mass and cannot separate one hand from another.

Figure 5.2.3 represents the convex hull and convexity defects which is an aspect of convex hull, useful for counting the number of fingers in a hand gesture recognition system. The convexity defect points (red dots) were detected by writing a Python script with the use of an OpenCV library. The code can be found in the appendix section of the document.

5.2.3 Track hand trajectories in the hand hygiene simulated video

Videos were recorded with one hand moving linearly along the x axis and a circular hand movement imitating the hand trajectory in the hand hygiene video. Then the contour and centroid was extracted for all video frames and a plot was constructed to illustrate the linear and circular hand movements. Similar steps were followed for tracking linear and circular hand movements with two hands in contact and thereby imitating the hand trajectory in hand hygiene pose, rub hands palm to palm. Figure 5.2.5 represents the linear and circular hand movements of one hand and Figure 5.2.6 represents the linear and circular movement with two hands in contact. Figure 5.2.7 represents the plot for circular hand movement recorded with the Leap Motion Controller and Camera at the same time by mounting them on an aluminium frame, separated by a distance of 5 cm (approx). The aluminium frame was built specifically for this project and is discussed in detail in the previous chapter.

Figure 5.2.8 represents the centroid plot for the hand hygiene stage, 'rub hands palm to palm'. The hand hygiene video-based dataset was constructed by mounting the camera over an aluminium frame that was specifically built for this project. The aluminium frame was placed over the laboratory's sink in order to imitate the real-time hand washing scenario.

Figure 5.2.9 represents the centroid plot for the hand hygiene stage, 'right palm over left dorsum'.

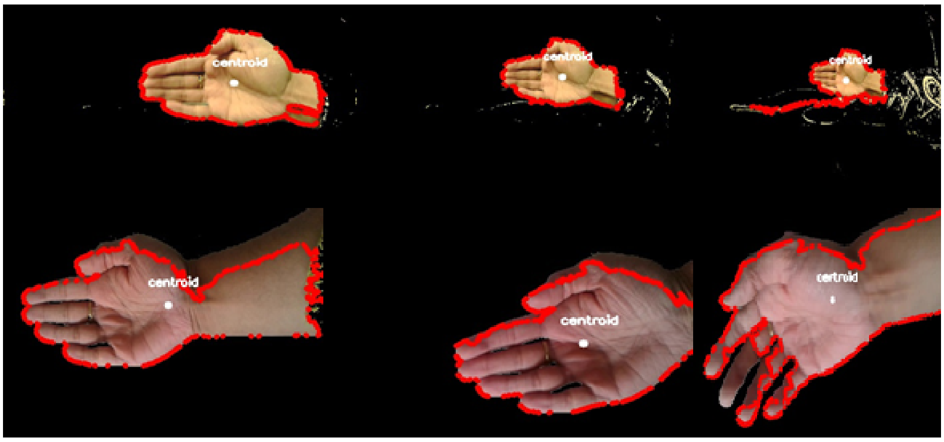

Figure 5.2.4: Selected video frames with contours and centroid for tracking linear hand movement

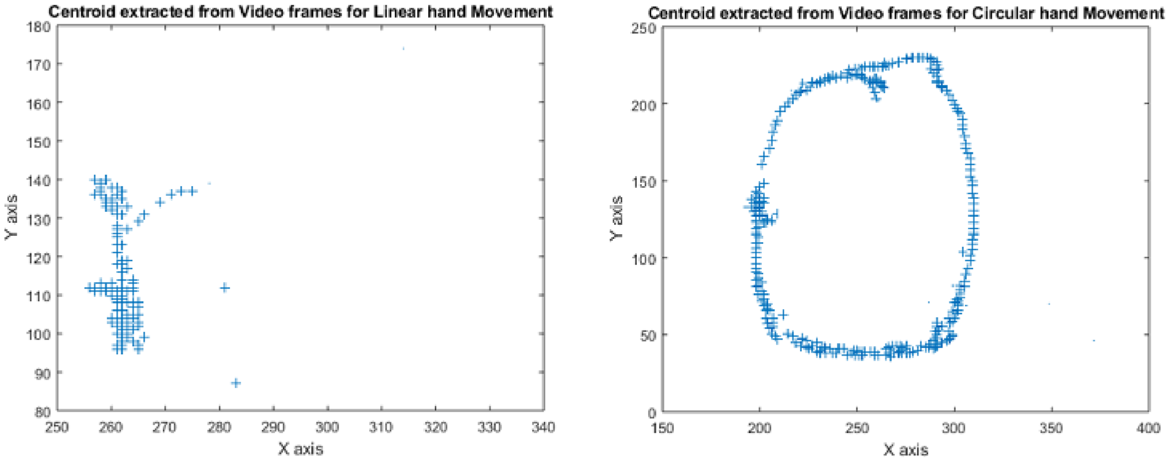

Figure 5.2.5: The centroid of a moving hand was extracted from the video frames demonstrating the linear and circular hand trajectory as it was observed from the hand hygiene video analysis

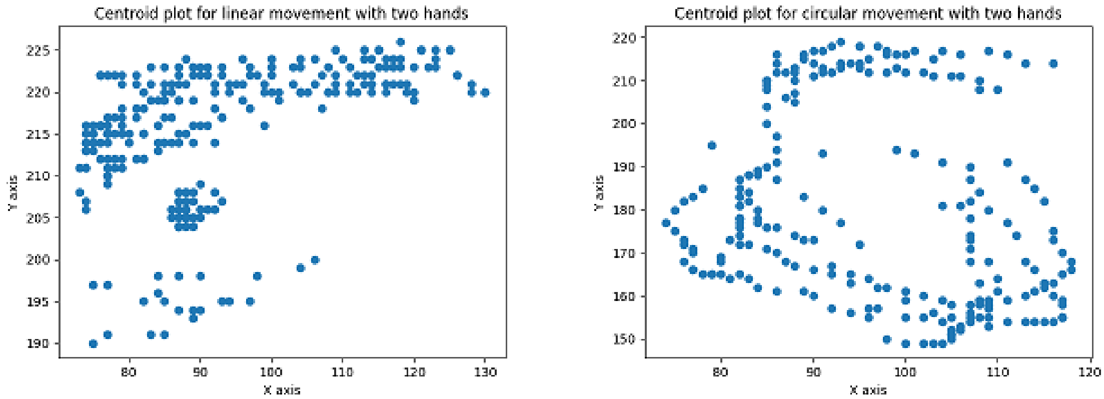

Figure 5.2.6: The centre of the mass(x,y values) of two hands in contact was extracted from the video frames for Linear(L) and circular(R) hand movements

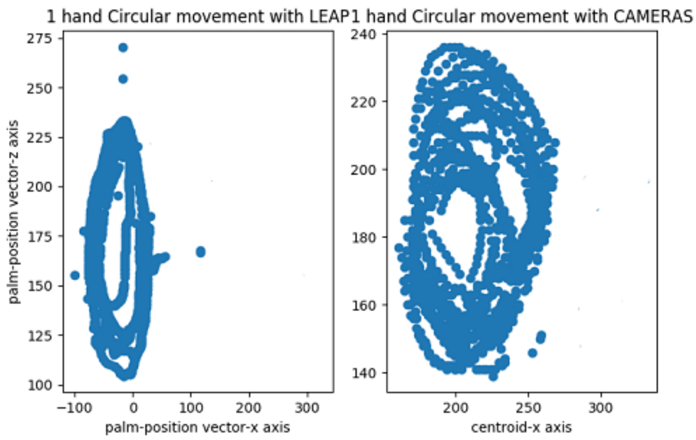

Figure 5.2.7: Circular hand movement plot from the Leap sensor and cameras; mounted over a frame.

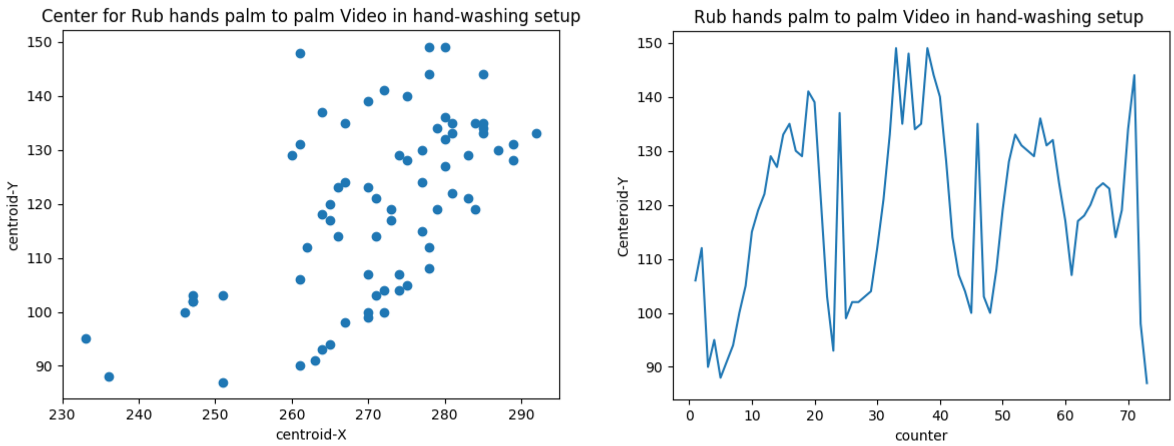

Figure 5.2.8: The centroid is extracted from the HH video-”rub hands palm to palm”

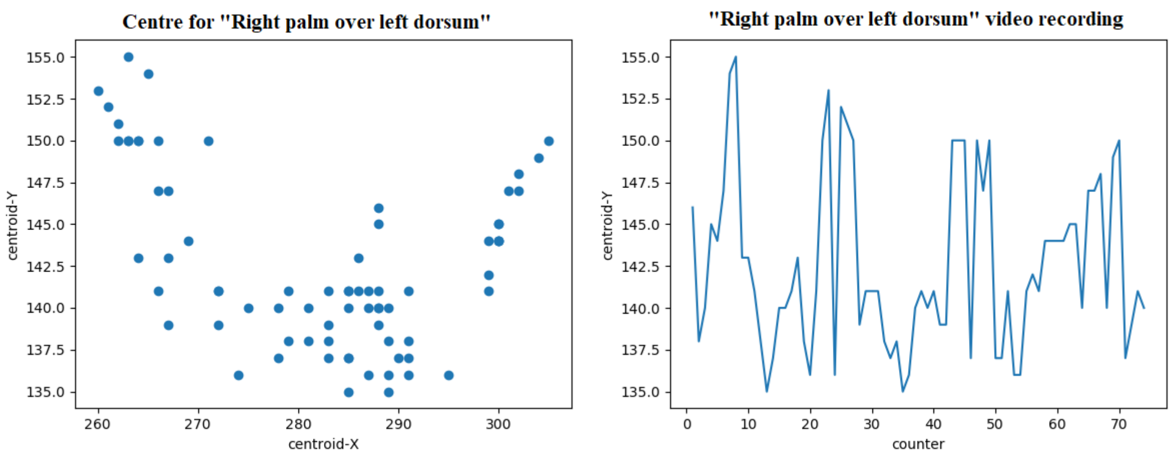

Figure 5.2.9: The centroid is extracted from the hand hygiene video-right palm over left dorsum.

5.3 Feature detection in computer vision

A feature is an attribute or a characteristic property of an object which is tracked.

3 types of image features in the field of computer vision are: [99]

- Corners (also known as Interest points)
- Edges
- Blobs (also known as region of interests)

A corner is an intersection of the two edges; it represents a point in which the directions of these two edges change. In other words, a corner represents the variation of intensity in an image.

Corners verses Edges

Corners are the locations where variations of intensity function f(x,y) in both X and Y are high. Both partial derivatives fx and fy are large. Edges are the locations where variation of f(x,y) in certain direction is high, while variation in the orthogonal direction is low. If an edge is oriented along Y, fx is large and fy is small. Figure 5.3.1 illustrates the difference between a corner and an edge and their x-y correspondence. Figure 5.3.2 differentiates between a flat region, an edge and a corner by demonstrating a change in the directions. For instance, a flat region does not change in all directions. An edge are those pixel values that do not change along an edge direction. A corner are those points that have significant variation in all the directions and therefore suitable for image classification [100].

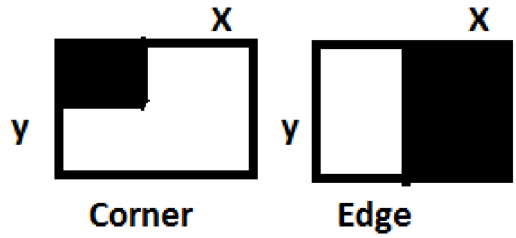

Figure 5.3.1: corners vs edges [100]

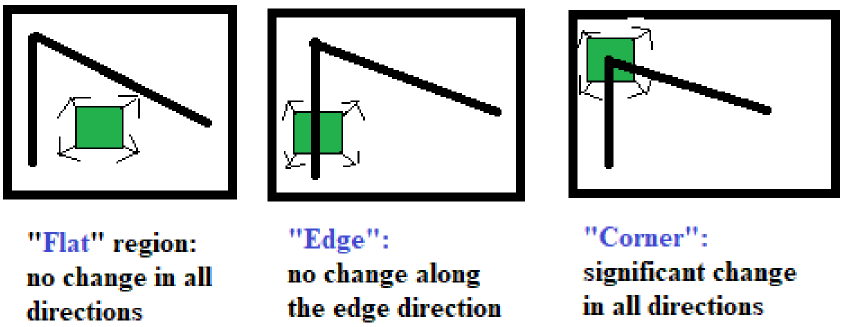

Figure 5.3.2: Difference between a flat region, an edge and a corner [100]

5.3.1 Properties of a strong feature

To build a robust hand feature detector based on camera images, one needs to understand and differentiate strong features from weak features. The general properties of a strong feature are: [100]

1. Features are local and accurate
2. Features are robust (to noise, blur, compression)
3. Distinct: Individual features can be matched to a large database.

4. Invariant: where a feature does not change in case of rotation, image scaling and lighting difference. For instance, a 'corner' identified as an 'edge' if the scale of an image changes is an example of a poor feature [100].

Corners are regarded as "good features" in comparison to edges as they are uniquely identifiable due to large variations in all directions.
There are various feature detection algorithms in the field of computer vision.

5.3.2 Harris corner detector

The basic idea in Harris corner detection algorithm is that one should be able to easily recognise the corner point by looking through a small window. Shifting a window in any direction should give a large change in the intensity. Change of intensity for the shift [u,v] is computed by: [101]

$$E(u,v) = \sum_{x,y} \underbrace{w(x,y)}_{\text{window function}} \, [\underbrace{I(x+u,y+v)}_{\text{shifted intensity}} - \underbrace{I(x,y)}_{\text{intensity}}]^2 \tag{5.7}$$

where u,v are the x,y coordinates of every pixel in the (3*3) window and I is the intensity value of the pixel.
E(u,v) is the sum squared difference of the pixel values in all directions. E(u,v) function is maximised for the corner detection by applying Taylor Expansion to the above equation. The final equation becomes

$$E(u,v) = \begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix} \tag{5.8}$$

where the summed matrix is renamed and put to M.
High level pseudo code for Harris corner detection algorithm is:

- Determine windows (small image patches) that produce large variations in intensity in both X and Y directions (gradients).
- For each such window found, a score R is computed.
- Threshold is applied to this score and important corners are selected.

Measure of a corner response(R) is calculated by

$$R = det(M) - k(trace(M))^2 \tag{5.9}$$

where

$$det(M) = \lambda_1 \lambda_2 \tag{5.10}$$

$$trace(M) = \lambda_1 + \lambda_2 \tag{5.11}$$

λ_1, λ_2 are the eigenvalues of M.

A window with a score R greater than a certain value is considered as a "corner". Figure 5.3.3 shows the threshold values for R.

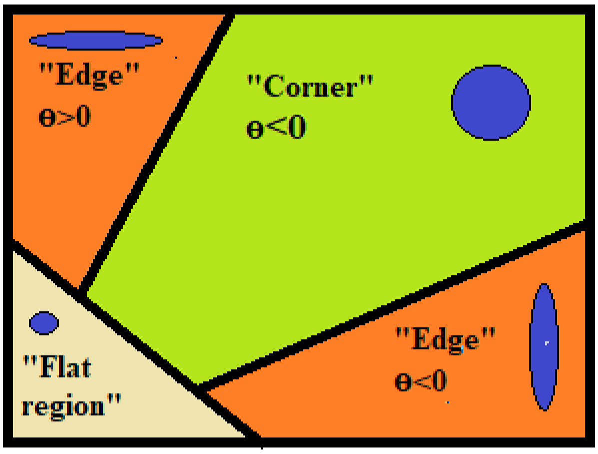

Figure 5.3.3: Value of R score determines if it is a corner [100]

5.3.3 Shi-Tomasi Detector

Shi-Tomasi modified Harris corner detection model by proposing different scoring criteria,

$$R = min(\lambda_1, \lambda_2) \tag{5.12}$$

They demonstrated in the paper that their scoring method provides better results compared to Harris corner detector. OpenCV has a function, cv2.goodFeaturesToTrack() that finds N strongest corners in the image by Shi-Tomasi method [102].

5.3.4 Scale Invariant Feature Transform Detector (SIFT)

SIFT is one of the most popular and widely used feature descriptors in the area of object recognition, motion tracking, feature matching and 3D construction.

Major steps involved in SIFT algorithm to generate image features are:

1. Feature point detection: An image is taken and blurred using Gaussian kernel; then the sum of second order derivatives of Gaussian is calculated which will locate edges and corners in an image.

2. Feature point localisation: Elimination of low contrast key points and edge points takes place so that the remainder is a list of strong features on the measures of their stability.

3. Orientation assignment: An orientation is assigned to each key point to achieve in-variance towards image rotation. Histogram of oriented gradient (HOG) is used with 36 bins covering 360 degree range of orientations.

4. Feature descriptor generation: At this point, we have a list of feature points which are described in terms of location, scale and orientation. The local coordinate system is constructed around the feature point that allows for significant levels of local shape distortion and change in illumination [103].

In OpenCV, SIFT is implemented using SIFT() and Sift detect() functions that find key points in an image.cv2.drawkeypoints() draw small circles over the locations of keypoints. I have tested all above-mentioned feature detectors for hand hygiene poses to extract useful hand features that can be fed to a classifier for further processing and classification.

5.3.5 Results for feature detection algorithms

Various feature detection algorithms such as Harris detector, Shi-tomasi detector and SIFT detector were discussed in the previous section.

These feature detectors are used to extract useful and strong features from an image or video recording that can be further fed to a machine learning classifier for image recognition and classification. In this work, these feature detectors are used to extract the features such as corners from an image of the hand hygiene pose, rub hands palm to palm.

Harris detector algorithm has produced many redundant features where edges were misinterpreted as corners and therefore it is not useful for the classification of various hand hygiene poses.

Shi-Tomasi detector algorithm has produced few, uniquely identifiable matches and yet has detected some edge points as corners. SIFT feature descriptor yields strong features, mainly corners yet higher in number in comparison to Shi-Tomasi and it is suitable for the hand pose classification.

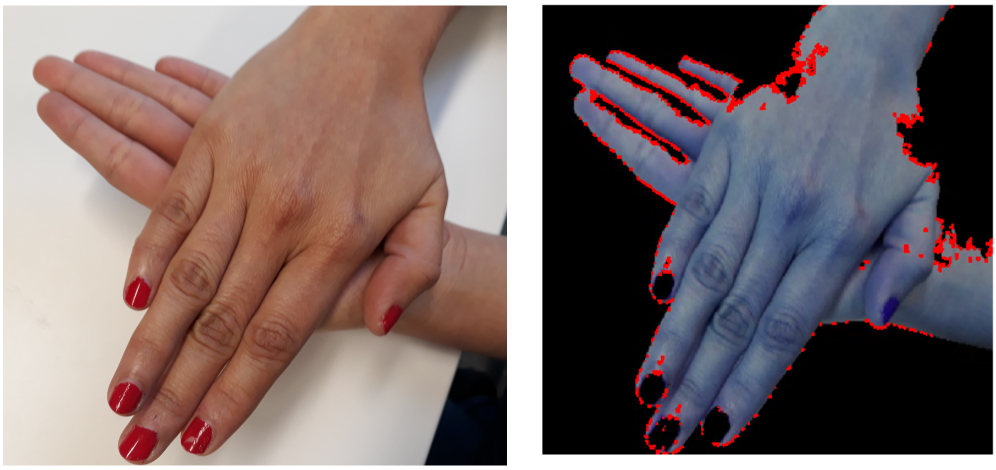

Figure 5.3.4: Original image and Harris features for the hand pose ”rub hands palm to palm”.

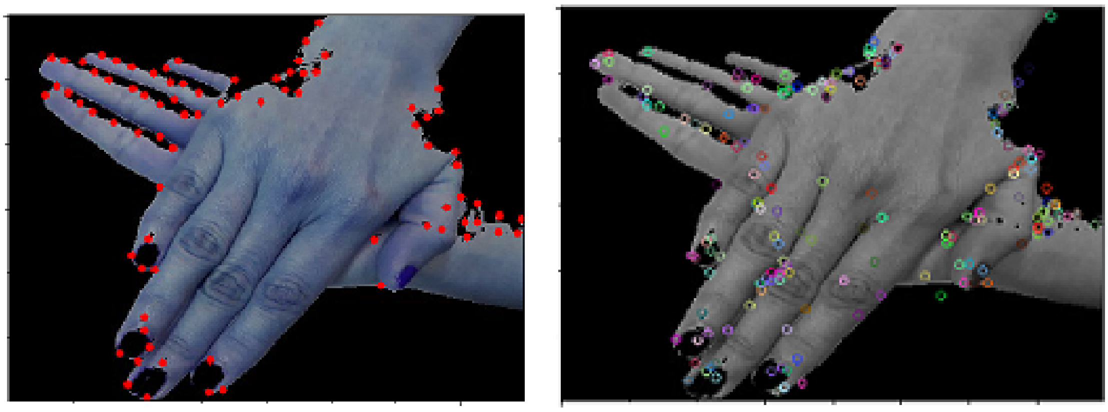

Figure 5.3.5: Shi-Tomasi features (left) and SIFT features (right) for a hand pose ”rub hands palm to palm”.

5.3.6 Extreme points in a hand image

Extreme points in an image, were detected by extracting maximum and minimum contour values from a numpy array in python environment. These x,y pixel values can be passed to a classifier for the purpose of hand pose classification. A python code can be found in the appendix section of the document.

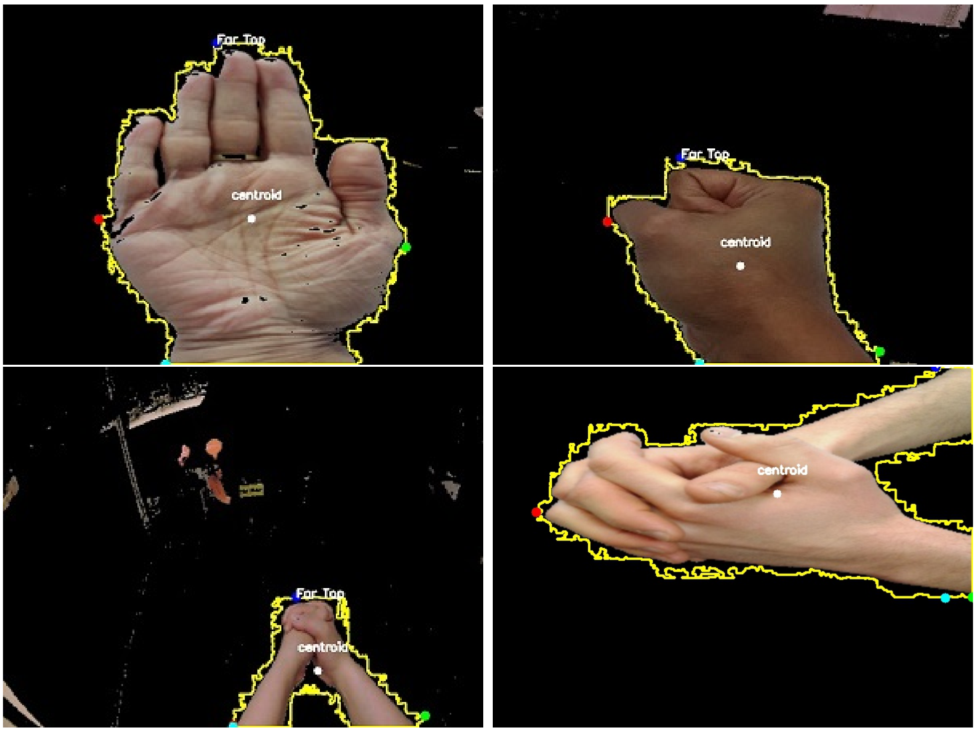

Figure 5.3.6: Extreme points detected for hands along with the centroid of the hand.

5.4 Discussion

1. **Skin extraction:** A variety of images with different background information were taken. Python based code was able to separate the foreground pixels (hands) from the background pixels. However, there was a problem when background resembled the skin color. For example: wood, leather, brown colored flooring were mis-classified as skin pixels and therefore such backgrounds were avoided in general. In some experiments, lighting conditions produced shadow and could not extract the hands in an accurate manner and so consistent lights were chosen. Figure 5.1.4 represents the hands extracted for a hand-hygiene pose.

2. **Contour and centroid detection:** Various one-hand gestures such as straight hand, closed hand, V-sign were tested to detect the overall contour and convex-hull attribute of the hand. With centroid, linear and circular hand movements were plotted for one hand and two hands in contact. The algorithm is useful in tracking hand movements. However, the key challenge is to identify and separate one hand from another. With a two-hand pose image, both hands were extracted as one single blob as shown in Figure 5.2.2. It is concluded that the segmentation of hands is difficult with cameras where as the Leap sensor can clearly identify left hand from right hand.

3. **Feature detection:** The Harris Feature detector algorithm is invariant to image rotation, meaning that the features do not change when an image is rotated. But they are not invariant to the image scale or intensity, meaning that the features are not reliable if the image is resized or the brightness differs. The aim is to detect the same interest points regardless of the image change. However, SIFT features are invariant to rotation, scale and illumination and therefore suited for image classification (Figure 5.3.4, Fig. 5.3.5).

4. **Extreme points in an image:** After contour detection, the extreme points (top, bottom, left and right x-y coordinates) were detected by extracting the minimum and maximum contour values from a numpy array. See appendix for the code. These extreme hand locations can be used for further hand-pose classification; such as an open hand vs closed hand as an example. (Figure 5.3.6)

Hand features	Can be detected by Cameras?	Can be detected by Leap?
Palm orientation	No	Yes(palm-normal vector)
Palm-shape	Yes (tracking contours)	Yes(hand-curvature function)
Finger spread	Yes (convex hull)	Yes(Euc. distance b/w fingers)
Hand trajectory	Yes(contours and centroid)	Yes(palm-position vector)
Frequency-time	Yes (video recording with timestamp)	Yes(frames per second)
2 hands tracking	Yes(centroid)	No(loss of data)

Table 5.1: Evaluation of the Leap and cameras w.r.t. hand features

5.5 Conclusion

Camera based images and videos were utilised to extract hands and track their movement with the help of OpenCV - an open source computer vision library for Python environment.

- Hands can be extracted from a source video having the background information with the help of a skin detection algorithm.
- Contours of a hand along with convex hull and convexity defect points were detected for an open hand. Convexity defect points are useful in counting the number of fingers and therefore these attributes of a hand contour can be further expanded for building a hand gesture recognition system.
- Extreme points on the hand were detected and feature detection algorithms were applied to hand hygiene pose- Rub hands palm to palm. These interest points gathered from the camera images can be combined with SVM to generate an image classifier.
- Linear and circular hand movements in the context of the hand washing application were tracked with the help of a centroid detection algorithm.

In this project, 3D gesture tracking device - the Leap Motion Controller and cameras were explored to evaluate their suitability in hand detection for static as well as dynamic hand movements. The key challenge faced with the Leap sensor was the " loss of data due to hand occlusion" and the difficulties that were faced with the cameras was the "segmentation of the hands". However, if combined, they may be used to create a robust hand gesture data set.

Chapter 6

Conclusions and Future work

The systematic and an accurate classification of various hand hygiene stages may require the use of machine- learning and deep learning models. In this work, 3D sensors and digital cameras were investigated for the purpose of tracking hands in the hand wash setup. The challenging task of segmentation and detection of an individual hand hygiene movement requires the collection of large amounts of data where users also have their personalised interpretation of hand hygiene guidelines and therefore wash their hands differently. In the next section and the following sections, the over all summary of the work along with the future work is discussed.

6.1 Conclusions

The overall project can be summarised into 4 phases.

1. Phase 1: Literature review

    (a) The importance of hand hygiene practice in the reduction of hospital acquired infections was researched.

    (b) There is low hand hygiene compliance towards WHO hand hygiene guidelines worldwide.

    (c) Professional roles can influence the compliance rate. For example, nurses have higher compliance rate in comparison to the physicians and other health care workers.

    (d) Hand gesture recognition system with the help of 3D sensors and cameras were researched, including the use of the Leap sensor and Microsoft Kinect for extracting the hand features such as finger tip position, hand centre, hand curvature for recognising the American sign language (ASL) gestures. Finger tracking and template matching for ASL gestures.

2. Phase 2: Feature Extraction for WHO hand hygiene stages was completed by analysing the professional hand washing videos. The following features were identified:
   (a) Palm orientation (facing in the same direction/facing away)
   (b) Palm shape (flat/curved)
   (c) Hand trajectory (linear/circular hand movement)
   (d) Finger spread(open hand/closed hand)
   (e) Frequency and amount of time taken to complete every HH stage.
3. Phase 3: The suitability of a 3D gesture tracking device, the Leap Motion Controller in tracking HH stages was assessed.
   (a) It is suited for tracking the single hand poses such as an open hand, a closed hand, linear hand movement, circular hand movement. It can measure the palm velocity and palm distance between the two hands.
   (b) The basic HH stage-"rub hands palm to palm" can be detected by counting the number of hands, calculating the palm distance between two hands, and measuring the palm velocity when hands are in contact.
   (c) The presence of a hand on the soap dispenser and a water tap in a hand washing setup can be detected with the help of the Leap sensor.
   (d) The drawbacks with the use of the Leap sensor are hand occlusion and the loss of data when two hands are in contact with each other.
4. Phase 4: The camera based approach for tracking HH stages was assessed.
   (a) Skin detection algorithms were used to extract skin pixels from the background with the help of a YCbCr color model.
   (b) Contour and centroid detection was shown for tracking linear and circular hand movement.
   (c) Convex hull and convexity defect points as an attribute of a contour were applied to an open hand. These points along with the extreme points can be used to count the number of fingers and gesture recognition in the real-time.

(d) Different feature detection algorithms such as Harris, Shi-Tomasi and SIFT were applied to a hand pose image-"rub hands Palm to Palm". These features are useful for the purpose of hand hygiene stage classification.

6.2 Future work

The hand hygiene video repository that was created by recording the hand washing movements of 30 participants is the most valuable output of the work done in this project. The data set is discussed in detail in Chapter 3.

The future work will be based on processing this data set and understanding the reasoning behind the personalised interpretation of the hand hygiene guidelines by the users. Centroid extraction for an individual's hand washing movement in a given amount of time is a useful feature that can be passed to a machine learning/deep learning classifier for the classification of various hand hygiene stages.

In addition, features extracted from the Leap software such as finger tip positions, palm position, palm velocities can be integrated with (x,y) pixel values generated from the digital camera such as centre of the hand, convex hull and convexity defect points for building a robust hand gesture recognition system.

Image API of the Leap sensor provides raw camera images for the frames recorded. They should be explored in the future work to conduct a comparison study between digital image data and the Leap sensor images.

As a future work, there are multiple routes that the project can take. However, the use of deep learning technology seems to be the most practical and a valuable direction with promising outputs. Information extracted from training the largely available hand gesture data set from: Kaggle; ImageNet can be applied to the smaller data set created in this project. Transfer of knowledge from the source domain to the target domain can solve the basic problem of insufficient data and can help in the classification of various hand gestures involved in the hand washing process.

Bibliography

[1] Bruno Zamborlin, Frédéric Bevilacqua, Marco Gillies, and Mark D'inverno. Fluid gesture interaction design: Applications of continuous recognition for the design of modern gestural interfaces. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 3, 01 2014.

[2] Abdul Butt, Alessandra Moschetti, Laura Fiorini, Paolo Dario, and Filippo Cavallo. *Wearable sensors for gesture analysis in smart healthcare applications*, pages 79–102. 06 2017.

[3] Sreejith Sasidharan Nair, Ramesh Hanumantappa, Shashidhar Gurushantswamy, Mohammed Asaduddin Siraj, and Pooja Raghunath. Knowledge , Attitude , and Practice of Hand Hygiene among Medical and Nursing Students at a Tertiary Health Care Centre in Raichur , India. *ISRN Preventive Medicine*, 2014.

[4] Susan B Barnes. User Friendly: A Short History of the Graphical User Interface. *Sacred Heart University Review*, 16(1), 1995.

[5] Liwen Xu. Multitouch and Gesture: A Literature Review of Multitouch and Gesture. *MultitouchAG*, 2014.

[6] I Bentley. Responsive environments. *Proceedings of the June 13-16, 1977, national computer conference on - AFIPS '77*, page 423, 1985.

[7] Ibm release simon pda. `https://history-computer.com/ModernComputer/Personal/Simon_of_IBM.html`, accessed: 2018.

[8] Nilofar E Chanda. Study of Touch Less Touch Screen Technology. *International journal of current engineering and scientific research*, (4):2394–0697, 2017.

[9] Jun Rekimoto. SmartSkin - An Infrastructure for Freehand Manipulation on Interactive Surfaces. *Proceedings of the International Conference on Human Factors in Computing Systems (CHI'02)*, pages 113–120, 2002.

[10] Soumi Paul, Subhadip Basu, and Mita Nasipuri. Microsoft Kinect in Gesture Recognition : A Short Review. *International Journal of Computer Technology and Applications*, 8(5):2071–2076, 2015.

[11] Frederic Magoules and Qinmeng Zou. A Novel Contactless Human Machine Interface Based on Machine Learning. *Proceedings - 2017 16th International Symposium on Distributed Computing and Applications to Business, Engineering and Science, DCABES 2017*, 2018-Septe(i):137–140, 2018.

[12] Riyad A. El-Laithy, Jidong Huang, and Michael Yeh. Study on the use of Microsoft Kinect for robotics applications. *Record - IEEE PLANS, Position Location and Navigation Symposium*, pages 1280–1288, 2012.

[13] Mohammed A. Hussein, Ahmed S. Ali, F. A. Elmisery, and R. Mostafa. Motion control of robot by using kinect sensor. *Research Journal of Applied Sciences, Engineering and Technology*, 8(11):1384–1388, 2014.

[14] Rajesh Kannan Megalingam, Nihil Saboo, Nitin Ajithkumar, Sreeram Unny, and Deepansh Menon. Kinect based gesture controlled Robotic arm: A research work at HuT Labs. *Proceedings of the 2013 IEEE International Conference in MOOC, Innovation and Technology in Education, MITE 2013*, (April 2015):294–299, 2013.

[15] Giulio Marin, Fabio Dominio, and Pietro Zanuttigh. Hand gesture recognition with jointly calibrated Leap Motion and depth sensor. *Multimedia Tools and Applications*, 2015.

[16] Stefanie T.L. Pöhlmann, Elaine F. Harkness, Christopher J. Taylor, and Susan M. Astley. Evaluation of Kinect 3D Sensor for Healthcare Imaging. *Journal of Medical and Biological Engineering*, 36(6):857–870, 2016.

[17] Erika Rovini, Carlo Maremmani, and Filippo Cavallo. How wearable sensors can support parkinson's disease diagnosis and treatment: A systematic review. *Frontiers in Neuroscience*, 11, 2017.

[18] G Ushaw, E Ziogas, J Eyre, and G Morgan. An Efficient Application of Gesture Recognition from a 2D Camera for Rehabilitation of Patients with Impaired Dexterity. pages 1–4, 2003.

[19] Wile Daryl, Ranjit Ranawaya, and Zelma Kiss. Smart watch accelerometry for analysis and diagnosis of tremor. *Journal of neuroscience methods*, 230, 2014.

[20] Who guidelines on hand hygiene in health care: a summary. `https://www.who.int/gpsc/5may/tools/who_guideline-handhygiene_summary.pdf`, 2009.

[21] Angela Revelas. Healthcare-associated infections: A public health problem. *Nigerian Medical Journal: Journal of the Nigeria Medical Association*, 2012.

[22] Purva Mathur. Hand hygiene Back to the basics of infection control Mathur P - Indian J Med Res. *Indian Journal Medical Research*, pages 611–620, 2011.

[23] Alessandro Cassini, Diamantis Plachouras, and et al. Burden of Six Health care-Associated Infections on European Population Health: Estimating Incidence-Based Disability-Adjusted Life Years through a Population Prevalence-Based Modelling Study. *PLoS Medicine*, 13(10):1–16, 2016.

[24] Pittet D Boyce JM. Guideline for Hand Hygiene in Health-Care Settings. Recommendations of the Healthcare Infection Control Practices Advisory Committee and the HICPAC/SHEA/APIC/IDSA Hand Hygiene Task Force. Society for Healthcare Epidemiology of America/Association for Professionals in Infection Control/Infectious Diseases Society of America. *PubMed,Medline*, 2002.

[25] Didier Pittet. Improving adherence to hand hygiene practice: A multidisciplinary approach. *Emerging Infectious Diseases*, 7(2):234–240, 2001.

[26] G. Kinsella, A. N. Thomas, and R. J. Taylor. Electronic surveillance of wall-mounted soap and alcohol gel dispensers in an intensive care unit. *Journal of Hospital Infection*, 66(1):34–39, 2007.

[27] S Sroka, P Gastmeier, and E Meyer. Impact of alcohol hand-rub use on meticillin- resistant Staphylococcus aureus : an analysis of the literature. *Journal of Hospital Infection*, 74(3):204–211, 2010.

[28] Daniel N Maxwell, Trish M Perl, and James B Cutrell. “The Art of War” in the Era of Coronavirus Disease 2019 (COVID-19). *Clinical Infectious Diseases*, 03 2020. ciaa229.

[29] Weekly epidemiological update-2 march 2021. "https://www.who.int/publications/m/item/weekly-epidemiological-update---2-march-2021", accessed: 2021.

[30] Yu Hu et al. W.Guan, Z.Ni. Clinical characteristics of coronavirus disease 2019 in china. *The New England journal of Medicine*, 2019.

[31] Cascella M, Rajnik M, and Cuomo A et al. Features, evaluation and treatment coronavirus (covid-19). "https://www.ncbi.nlm.nih.gov/books/NBK554776/", accessed: 2020.

[32] Coronavirus disease 2019(covid-19) situation report-120. "https://www.who.int/docs/default-source/coronaviruse/situation-reports/20200519-covid-19-sitrep-120.pdf?sfvrsn=515cabfb_2", accessed: 2020.

[33] How to protect yourself. "https://www.cdc.gov/coronavirus/2019-ncov/prepare/prevention.html", accessed: 2020.

[34] Manoj Ramachandran Daud Tai Shan Chou, Pramod Achan. The World Health Organization'5 Moments of Hand Hygiene': The scientific foundation. *The Bone and Joint Journal*, 94:441–5, 2012.

[35] Melissa A Ward, Marin L Schweizer, Philip M Polgreen, Kalpana Gupta, Heather S Reisinger, and Eli N Perencevich. Automated and electronically assisted hand hygiene monitoring systems : A systematic review. *American Journal of Infection Control*, 42(5):472–478, 2014.

[36] Humberto Guanche Garcell, Ariadna Villanueva Arias, Fernando Ramírez Miranda, Reynol Rubiera Jimenez, Ramón N Alfonso Serrano, and Humberto Guanche Garcell. Direct observation of hand hygiene can show differences in staff compliance : Do we need to evaluate the accuracy for patient safety ? *Qatar Medical Journal*, 2017:1–6, 2017.

[37] Sung Ching Pan, Kuei Lien Tien, I. Chen Hung, Yu Jiun Lin, Wang Huei Sheng, Ming Jiuh Wang, Shan Chwen Chang, Calvin M. Kunin, and Yee Chun Chen. Compliance of Health Care Workers with Hand Hygiene Practices: Independent Advantages of Overt and Covert Observers. *PLoS ONE*, 8(1):1–7, 2013.

[38] J. M. Al Salman, S. Hani, N. de Marcellis-Warin, and Sister Fatima Isa. Effectiveness of an electronic hand hygiene monitoring system on healthcare workers' compliance to guidelines. *Journal of Infection and Public Health*, 8(2):117–126, 2015.

[39] Lisa L. Pineles, Daniel J. Morgan, Heather M. Limper, Stephen G. Weber, Kerri A. Thom, Eli N. Perencevich, Anthony D. Harris, and Emily Landon. Accuracy of a radiofrequency identification (RFID) badge system to monitor hand hygiene behavior during routine clinical activities. *American Journal of Infection Control*, 42(2):144–147, 2014.

[40] John M. Boyce. Electronic monitoring in combination with direct observation as a means to significantly improve hand hygiene compliance. *American Journal of Infection Control*, 45(5):528–535, 2017.

[41] Md Abu Sayeed Mondol and John A. Stankovic. Harmony: A Hand Wash Monitoring and Reminder System using Smart Watches. *Proceedings of the 12th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, 2015.

[42] P.M. Polygreen, C.S. Hlady, M.A. Severson, A.M. Segre, and T Herman. Method for Automated Monitoring of Hand Hygiene Adherence without Radio-Frequency Identification. *Infect Control Hosp Epidemiol.*, 31(12):1294–1297, 2011.

[43] Hang Thi Phan, Hang Thi Thuy Tran, Hanh Thi My Tran, Anh Pham Phuong Dinh, Ha Thanh Ngo, Jenny Theorell-Haglow, and Christopher J. Gordon. An educational intervention to improve hand hygiene compliance in Vietnam. *BMC Infectious Diseases*, 18(1):1–6, 2018.

[44] M. Musu, A. Lai, N. M. Mereu, M. Galletta, M. Campagna, M. Tidore, M. F. Piazza, L. Spada, M. V. Massidda, S. Colombo, P. Mura, and R. C. Coppola. Assessing hand hygiene compliance among health care workers in six Intensive Care Units. *Journal of Preventive Medicine and Hygiene*, 58(3):E231–E237, 2017.

[45] J. Randle, M. Clarke, and J. Storr. Hand hygiene compliance in healthcare workers. *Journal of Hospital Infection*, 64(3):205–209, 2006.

[46] Vicki Erasmus, Thea J Daha, Hans Brug, Jan Hendrik Richardus, Myra D Behrendt, Margreet C Vos, and Ed F Van Beeck. Systematic Review of Studies on Compliance with Hand Hygiene Guidelines in Hospital Care. *infection control and hospital epidemiology*, 31(3), 2010.

[47] Farinaz Farhoudi, Anahita Sanaei Dashti, Minoo Hoshangi Davani, Nadiyeh Ghalebi, Golnar Sajadi, and Raziyeh Taghizadeh. Impact of WHO Hand Hygiene Improvement

Program Implementation: A Quasi-Experimental Trial. *BioMed Research International*, 2016, 2016.

[48] Utility of electronic hand hygiene counting devices for measuring physicians' hand hygiene adherence applied to outpatient settings. *American Journal of Infection Control*, 44(12):1481–1485, 2016.

[49] Hand washing practice among health care workers in a teaching hospital. *Journal of Nepal Health Research Council*, 11(23):1–5, 2013.

[50] Sharifah Mohidin, Sana Al Eidarous, Laila Al Tabsh, and Abeer Al Hazmi. Observational Study on Compliance to Hand Hygiene among Health Care Workers in KAUH. *1st Annual Worldwide Nursing Conference*, 2013.

[51] Mais Yasen and Shaidah Jusoh. A systematic review on hand gesture recognition techniques, challenges and applications. *PeerJ Computer Science*, 5, 2019.

[52] Chuck Pheatt and Andrew Wayman. Using the xbox kinect™ sensor for gesture recognition. *Journal of Computing Sciences in Colleges*, 28:226–227, 2013.

[53] Leap motion controller- an api overview. `https://developer-archive.leapmotion.com/documentation/python/devguide/Leap_Overview.html`.

[54] Prakhar Mohan, Shreya Srivastava, Garvita Tiwari, and Rahul Kala. Region Extraction and Static Gesture Recognition. *2015 Eighth International Conference on Contemporary Computing (IC3)*, pages 144–149, 2015.

[55] Rung-huei Liang, Ming Ouhyoung, and Information Management. A Real-time Continuous Gesture Recognition System for Sign Language Dept . of Computer Science and Information Engineering , National Taiwan University , Taipei 106. *International Conference on Automatic Face and Gesture Recognition*, pages 558–567, 1998.

[56] Parvin Asadzadeh, Lars Kulik, and Egemen Tanin. Real-time Gesture Recognition Using RFID Technology. *Personal and Ubiquitous Computing*, 16:225–234, 2012.

[57] J. Hoey. Tracking using Flocks of Features, with Application to Assisted Handwashing. *Procedings of the British Machine Vision Conference 2006*, pages 38.1–38.10, 2006.

[58] Marouane Hamda and Abdelhak Mahmoudi. Hand Gesture Recognition Using Kinect's Geometric and HOG Features. *Proceedings of the 2nd international Conference on Big Data, Cloud and Applications-BDCA'17*, pages 1–5, 2017.

[59] Fabio Dominio, Mauro Donadeo, Giulio Marin, Pietro Zanuttigh, and Guido Maria Cortelazzo. Hand gesture recognition with depth data. *Proceedings of the 4th ACM/IEEE international workshop on Analysis and retrieval of tracked events and motion in imagery stream (ARTEMIS)*, pages 9–16, 2013.

[60] Lin Shao. Hand movement and gesture recognition using Leap Motion Controller. *Stanford.Edu*, pages 2–6, 2015.

[61] Haiyang Jin, Qing Chen, Zhixian Chen, Ying Hu, and Jianwei Zhang. Multi-LeapMotion sensor based demonstration for robotic refine tabletop object manipulation task. *CAAI Transactions on Intelligence Technology*, 1(1):104–113, 2016.

[62] David Fernández Llorca, Ignacio Parra, Miguel Ángel Sotelo, and Gerard Lacey. A vision-based system for automatic hand washing quality assessment. *Machine Vision and Applications*, 22(2):219–234, 2011.

[63] Baiqiang Xia, Rozenn Dahyot, Jonathan Ruttle, Darren Caulfield, and Gerard Lacey. Hand Hygiene Poses Recognition with RGB-D Videos. *Proceedings of the 17th Irish Machine Vision and Image Processing conference*, pages 43–50, 2015.

[64] Tahir Khan. Hand Gesture Recognition based on Digital Image Processing using MATLAB. *International Journal of Scientific & Engineering Research*, 6(9):338–346, 2015.

[65] Shany Jophin. Gesture Based Interface Using Motion and Image Comparison. *International Journal of Advanced Information Technology*, 2(3):37–46, 2012.

[66] Reza Azad, Babak Azad, and Iman Tavakoli Kazerooni. Real-time and robust method for hand gesture recognition system Based on cross-correlation coefficient. *Advances in computer science: an international journal,*, 2(5):121–125, 2013.

[67] P.R.V. Chowdary, M.N. Babu, T.V. Subbareddy, B.M. Reddy, and V. Elamaran. Image processing algorithms for gesture recognition using MATLAB. *2014 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)*, (978):1511–1514, 2014.

[68] Zahangir Alom, Tarek M Taha, Chris Yakopcic, Stefan Westberg, Paheding Sidike, Mst Shamima Nasrin, Mahmudul Hasan, Brian C Van Essen, Abdul A S Awwal, and Vijayan K Asari. A State-of-the-Art Survey on Deep Learning Theory and Architectures. *MDPI-Electronics*, pages 1–67, 2019.

[69] Xin Jia. Image recognition method based on deep learning. *29th Chinese Control And Decision Conference (CCDC)*, pages 4730–4735, 2017.

[70] Yann Lecun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521:436–444, 2015.

[71] What is deep learning?-by mathworks. `https://uk.mathworks.com/discovery/deep-learning.html`, accessed: 2019.

[72] Rikiya Yamashita, Mizuho Nishio, Richard Kinh, Gian Do, and Kaori Togashi. Convolutional neural networks : an overview and application in radiology. *Insights into Imaging*, pages 611–629, 2018.

[73] Satoshi Iizuka. Let there be Color !: Joint End-to-end Learning of Global and Local Image Priors for Automatic Image Colorization with Simultaneous Classification. *ACM transactions*, 35(4):1–11, 2016.

[74] Andrej Karpathy. Deep Visual-Semantic Alignments for Generating Image Descriptions. *Conference on Computer Vision and Pattern Recognition-IEEE Explore*, 2015.

[75] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, Andrew Zisserman, and C V Dec. Reading Text in the Wild with Convolutional Neural Networks. *Computer Vision and Pattern Recognition*, 2014.

[76] Andrew Owens, Antonio Torralba, Phillip Isola, Edward H Adelson, Josh Mcdermott, William T Freeman, and U C Berkeley. Visually Indicated Sounds. *Computer vision and Pattern Recognition*, 2016.

[77] Eli Shechtman. Deep Photo Style Transfer. *Computer vision and Pattern Recognition*, 2017.

[78] Keiron Teilo O Shea. An Introduction to Convolutional Neural Networks An Introduction to Convolutional Neural Networks. *arXiv preprint*, (December), 2015.

[79] Convolutional neural network. `https://uk.mathworks.com/discovery/convolutional-neural-network-matlab.html`, accessed: 2019.

[80] Diego R.Faria Mahbub Hussain, Jordan J.Bird. A Study on CNN Transfer Learning for Image Classification. *Advances in Computational Intelligence systems*, pages 191–202, 2018.

[81] Chuanqi Tan and Tao Kong Fuchun Sun. A Survey on Deep Transfer Learning. *27th International Conference on Artificial Neural Networks*, 2018.

[82] Yu-Chuan Su, Tzu-Hsuan Chiu, Chun-Yen Yeh, Hsin-Fu Huang, and Winston H. Hsu. Transfer Learning for Video Recognition with Scarce Training Data for Deep Convolutional Neural Network. *Computer Vision and Pattern Recognition*, 2015.

[83] Onur Asan and Enid Montague. Using video-based observation research methods in primary care health encounters to evaluate complex interactions. *Inform Prim Care*, 21(4):161–170, 2015.

[84] Ahmed Elgammal, Crystal Muang, and Dunxu Hu. Skin Detection - a Short Tutorial. *Encyclopaedia of Biometrics*, pages 1–10, 2009.

[85] M Aznaveh, H Mirzaei, E Roshan, and M Saraee. A new and improved skin detection method using mixed color space. *Human-Computer Systems*, pages 471–480, 2009.

[86] Sinan A Naji, Roziati Zainuddin, and Hamid A Jalab. Skin segmentation based on multi pixel color clustering models. *Digital Signal Processing*, 22(6):933–940, 2012.

[87] Faliang Chang, Zhiqiang Ma, and Wei Tian. A Region-Based Skin Color Detection Algorithm. *The 11th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 417–418, 2007.

[88] Amit Kumar and Shivani Malhotra. Pixel-Based Skin Color Classifier : A Review Pixel-Based Skin Color Classifier : A Review. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, (April 2016), 2015.

[89] Vladimir Vezhnevets. A Survey on Pixel-Based Skin Color Detection Techniques. *International Conference Graphicon*, 2003.

[90] Michael J. Jones and James M. Rehg. Statistical color models with application to skin detection. *Int. J. Comput. Vision*, 46(1), 2002.

[91] S. Kolkur, D. Kalbande, P. Shimpi, C. Bapat, and J. Jatakia. Human Skin Detection Using RGB, HSV and YCbCr Color Models. *Advances in Intelligent Systems Research*, 137:324–332, 2017.

[92] Jorge Basilio, Gualberto Torres, Gabriel Sanchez-Perez, L. Medina, and Hector Perez-Meana. Explicit image detection using ycbcr space color model as skin detection. pages 123–128, 2011.

[93] Image processing with python. https://datacarpentry.org/image-processing/09-contours/, accessed: 2018.

[94] Contour features-opencv documentation. https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_contours/py_contour_features/py_contour_features.html, accessed: 2018.

[95] Advanced contour properties. "https://gurus.pyimagesearch.com/lesson-sample-advanced-contour-properties/#", accessed: 2018.

[96] Guobo Xie and Wen Lu. Image Edge Detection Based On Opencv. *International Journal of Electronics and Electrical Engineering*, 1(2):104–106, 2013.

[97] Xhensila Poda and Olti Qirici. Shape detection and classification using OpenCV and Arduino Uno. *Proceedings of the 3rd International Conference RTA-CSIT*, pages 128–136, 2018.

[98] Sergey O Bochkarev, Igor B Litus, and Natalia S Kravchenko. Irregular Objects.Shape Detection and Characteristic Sizes. *Proceedings of the 1st International Workshop on Radio Electronics and Information Technologies*, 1(2), 2017.

[99] Harris corner detector. "https://docs.opencv.org/3.4/d4/d7d/tutorial_harris_detector.html", accessed: 2019.

[100] Saad J Bedros. Interest point detection. http://me.umn.edu/courses/me5286/vision/Notes/2015/ME5286-Lecture8.pdf, accessed: 2019.

[101] Christopher G. Harris and Mike Stephens. A combined corner and edge detector. *Alvey Vision Conference*, 1988.

[102] Jianbo Shi and Carlo Tomasi. Good features to track. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition.*, 600, 2000.

[103] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, November 2004.

[104] Rolf Molich and Jakob Nielsen. Improving a Human-Computer Dialogue. *Communications of the ACM*, 33(3), 1990.

[105] Emily Gonzalez-holland, Daphne Whitmer, Larry Moralez, and Mustapha Mouloua. Examination of the use of Nielsen's 10 usability heuristics & outlooks for the future. *Proceedings of the Human Factors and Ergonomics Society*, pages 1472–1475, 2017.

[106] Nicu Sebe. Multimodal interfaces: Challenges and perspectives. *Journal of Ambient Intelligence and Smart Environments*, 2014.

[107] Alejandro Correa Bahnsen. Machine learning algorithms explained–k-means clustering. `https://blog.easysol.net/machine-learning-algorithms-3/`, accessed: 2019.

[108] Manu Jeevan. The simplest way to explain k-means algorithm. `https://bigdata-madesimple.com/possibly-the-simplest-way-to-explain-k-means-algorithm/`, accessed: 2020.

[109] D. Lee, Kiyeol Baek, J. Lee, and H. Lim. A development of virtual reality game utilizing kinect, oculus rift and smartphone. *International Journal of Applied Engineering Research*, 11:829–833, 03 2016.

[110] Cory Maklin. K nearest neighbor algorithm in python. `"https://towardsdatascience.com/k-nearest-neighbor-python-2fccc47d2a55"`, accessed:2019.

[111] Global surveillance for covid-19 disease caused by human infection with the 2019 novel coronavirus. `"https://www.who.int/publications-detail/global-surveillance-for-human-infection-with-novel-coronavirus-(2019-ncov)"`, accessed: 2020.

APPENDIX

LINKS for hand-washing videos

https://www.youtube.com/watch?v=3PmVJQUCm4E

https://www.youtube.com/watch?v=zCVu_1d9AJ8

https://www.youtube.com/watch?v=ZOhBD5xi2mA

https://www.youtube.com/watch?v=bAwSOUslEDs

https://www.youtube.com/watch?v=AbAe69sKlSw

https://www.youtube.com/watch?v=mWe51EKbewk

https://www.youtube.com/watch?v=KusIuq8wu_0

https://www.youtube.com/watch?v=i_FsOQrtu90

https://www.youtube.com/watch?v=a9CMtzymZTg

https://www.youtube.com/watch?v=yJeDncdWdb8

```
# -*- coding: utf-8 -*-
"""Created on Fri Jun 15 09:34:00 2018
@author:Rashmi Bakshi
This piece of code utilizes Leap's palm position vector
to track Linear and Circular hand movement.
"""
import os, sys, inspect, thread, time
src_dir = os.path.dirname(inspect.getfile(inspect.currentframe()))
arch_dir = '../lib/x64' if sys.maxsize > 2**32
else 'C:/Leap/
LeapDeveloperKit/LeapSDK/lib/x86'
sys.path.insert(0, os.path.abspath(os.path.join(src_dir, arch_dir)))
import Leap
import cv2
import numpy as np
import ctypes
import base64
from matplotlib import pyplot as pl
from PIL import Image
import matplotlib.image as mp
from subprocess import *
```

```
import decimal
import time
class SampleListener(Leap.Listener):

    def on_connect(self, controller):
        print "Connected"
        def on_frame(self, controller):
            controller.set_policy(Leap.Controller.POLICY_IMAGES)
        frame = controller.frame()
            f1=open("L2-Side-Linear2.csv",'a')
        f2=open("R2-Side-Linear2.csv",'a')
        t0=time.time()
        for hand in frame.hands:

            if hand.is_left:
                a=str(hand.palm_position)
                mylist=a.split(',')
                x1=mylist[0]
                y1=mylist[1]
                z1=mylist[2]
                f1.write(x1+'\t'+y1+"\t"+z1+"\n")
                print a + " left hand"

            if hand.is_right:
                b=str(hand.palm_position)
                list1=b.split(',')
                x2=list1[0]
                y2=list1[1]
                z2=list1[2]
                f2.write(x2+"\t"+y2+"\t"+z2+"\n")
                print b + " right hand"
def main():
    listener = SampleListener()
```

```
    controller = Leap.Controller()
    controller.add_listener(listener)

    # Keep this process running until Enter is pressed
    print "Press Enter to quit..."
    try:
        sys.stdin.readline()
    except KeyboardInterrupt:
        pass
    finally:
        # Remove the sample listener when done
        controller.remove_listener(listener)

if __name__ == "__main__":
    main()
```

```
"""Created on Fri Jun 15 09:34:00 2018
@author:Rashmi Bakshi
This piece of code utilizes Leap's palm normal vector,
position vector
and grab strength function
to track Hand hygiene- palm to palm stage.
"""
import os, sys, inspect, thread, time
src_dir = os.path.dirname(inspect.getfile(inspect.currentframe()))
arch_dir = '../lib/x64' if sys.maxsize > 2**32
else
'C:/Leap/LeapDeveloperKit/LeapSDK/lib/x86'
sys.path.insert(0, os.path.abspath(os.path.join(src_dir, arch_dir)))
import Leap
import cv2
import numpy as np
import ctypes
import base64
from matplotlib import pyplot as pl
from PIL import Image
import matplotlib.image as mp
from subprocess import *
import decimal

class SampleListener(Leap.Listener):

    def on_connect(self, controller):
        print "Connected"

    def on_frame(self, controller):
        controller.set_policy(Leap.Controller.
        POLICY_IMAGES)
```

```
frame = controller.frame()
global gxL,gxR,Lgx,Lgy,Lgz,Rgx,Rgy,Rgz,veloc

for hand in frame.hands:

    if hand.is_left:
        strengthL=hand.grab_strength
        if strengthL <1.0:
            palmL= str(hand.palm_normal)
            #time.sleep(1)
            mylist=palmL.split(',')
            xval=mylist[0]
            xval=xval.split('(')
            xL=float(xval[1])
            gxL=xL

            palmPosL= str(hand.palm_position)
            #time.sleep(1)
            mylist3=palmPosL.split(',')
            x=mylist3[0]
            y=mylist3[1]
            z =mylist3[2]
            z=z.split(')')
            x=x.split('(')
            Lgx=float(x[1])
            Lgy=float(y)
            Lgz=float(z[0])

    if hand.is_right:
        strengthR=hand.grab_strength
        hand_speed = str(hand.palm_velocity)
        mylist=hand_speed.split(',')
        v=mylist[0]
```

```
        v=v.split('(')

        veloc=float(v[1])
        if strengthR<1.0:

            #print "flat right hand"
            palmR=str(hand.palm_normal)
            #time.sleep(1)
            mylist2=palmR.split(',')
            xvalR=mylist2[0]
            xvalR=xvalR.split('(')
            xR=float(xvalR[1])

            gxR=xR
            palmPosR=str(hand.palm_position)
            #time.sleep(1)

            mylist4=palmPosR.split(',')
            xr=mylist4[0]
            yr=mylist4[1]
            zr=mylist4[2]
            zr=zr.split(')')
            xr=xr.split('(')
            Rgx=float(xr[1])
            Rgy=float(yr)
            Rgz=float(zr[0])

        #else:
            #print"curved right "

#print gxL,gxR

s=gxL+gxR  # sum of x values of left hand
```

```
#and right hand palm normal
    #print gxR
    #s=10

#print s

#strength=strengthR+strengthL;
if strengthR<0.5 or strengthL<0.5:
    print "Flat hands"
else:
    print"Please keep palms flat"
if -0.4 < s<0.4:
    #print " hands are facing each other"
    print "Hands facing"

else:
    print "Please ensure hands are facing"
dix=(Rgx-Lgx)*(Rgx-Lgx)
diy=(Rgy-Lgy)*(Rgy-Lgy)
diz=(Rgz-Lgz)*(Rgz-Lgz)
su=dix+diy+diz
#print su
result=np.sqrt(su)
#print result
#print dix, diy, diz
if result<50:
    print "Hands are in touch"
else:
    print "Hands are apart."
ab=abs(veloc)
if (ab>150):
    print "hands are moving."
else:
```

```
            print "hands are not moving"

def main():
    listener = SampleListener()
    controller = Leap.Controller()
    controller.add_listener(listener)

    # Keep this process running until Enter is pressed
    print "Press Enter to quit..."
    try:
        sys.stdin.readline()
    except KeyboardInterrupt:
        pass
    finally:
        # Remove the sample listener when done
        controller.remove_listener(listener)

if __name__ == "__main__":
    main()
```

```python
# -*- coding: utf-8 -*-
"""
Created on Fri Jun 15 09:34:00 2018

@author: Rashmi
This python script calculates the Palm distance
between the hand and the Tap.
When the hand is on the tap,
the distance is reduced and a
threshold value is selected
to determine if the hand is in
contact with the Tap.
The similar script is written
to determine if hand is on the
soap Dispenser
"""

import os, sys, inspect, thread, time
src_dir = os.path.dirname(inspect.getfile
(inspect.currentframe()))
arch_dir = '../lib/x64' if sys.maxsize > 2**32
else 'C:/Leap/LeapDeveloperKit/LeapSDK/lib/x86'
sys.path.insert(0, os.path.abspath(os.path.join(src_dir, arch_dir)))
import Leap
import cv2
import numpy as np
import ctypes
import base64
from matplotlib import pyplot as pl
from PIL import Image
import matplotlib.image as mp
from subprocess import *
```

```
import decimal
import time
import math
class SampleListener(Leap.Listener):

    def on_connect(self, controller):
        print "Connected"



    def on_frame(self, controller):

        controller.set_policy(Leap.Controller
        .POLICY_IMAGES)
        frame = controller.frame()
        f1=open("DistanceTapRight.csv",'a')



        myTapX=103.78
        myTapY=264.89
        myTapZ=147.80

        for hand in frame.hands:

            if hand.is_right:

                b=str(hand.palm_position)



                list1=b.split(',')
                x1=list1[0]
                y1=list1[1]
                z1=list1[2]
```

```
                    x1=x1.split('(')
                    z1=z1.split(')')
                    xnew=float(x1[1])
                    y1=float(y1)
                    z1=float(z1[0])
distance = math.sqrt( ((myTapX-xnew)**2)+
((myTapY-y1)**2) +
((myTapZ-z1)**2) )
                    f1.write(str(distance)+ "\n")
                    if distance>30:
                        print"Hand is away from the Water Tap"
                    else:
                        print"Hand has reached to the Water Tap"
def main():
    listener = SampleListener()
    controller = Leap.Controller()
    controller.add_listener(listener)

    # Keep this process running until Enter is pressed
    print "Press Enter to quit..."
    try:
        sys.stdin.readline()
    except KeyboardInterrupt:
        pass
    finally:
        # Remove the sample listener when done
        controller.remove_listener(listener)

if __name__ == "__main__":
    main()

# -*- coding: utf-8 -*-
"""Created on Wed Sep 26 16:54:46 2018
@author: Rashmi
```

```
"""
"""I'm extracting skin pixels from the list of images..
\\ converting rgb image to ycrcb color space,
applying a mask
and Gaussian blur to remove the extra noise.
"""
import numpy as np
import glob
import cv2
import os, os.path
image_list = []
min_YCrCb = np.array([0,133,77],dtype="uint8")
max_YCrCb = np.array([255,179,127],dtype="uint8")
i=0
for filename in glob.glob('vid2frames/*.jpg'):
    im=cv2.imread(filename)
    im=cv2.resize(im,(400,300))
    image_list.append(im)

for a in image_list:
    converted = cv2.cvtColor(a, cv2.COLOR_BGR2YCR_CB)
    mask=cv2.inRange(converted,min_YCrCb,
    max_YCrCb)

    skinMask = cv2.GaussianBlur(mask, (3, 3), 0)
    skinMask = cv2.bitwise_and(a, a, mask = mask)

# show the skin in the image along with the mask
    cv2.imshow("images", np.hstack([a, skinMask]))
    cv2.imwrite('extracted'+str(i)+'.jpg',
    skinMask)
    i+=1
    #cv2.imshow('img',image_list[0])
```

```
# if the 'q' key is pressed, stop the loop
    k = cv2.waitKey(0)

cv2.destroyAllWindows()
```

```
"""Created on Wed Sep 26 16:54:46 2018
@author: Rashmi
Reading image files from the folder;
and finding the largest contour
and its centre for each image
for the purpose of tracking
linear and circular hand movement
"""
from PIL import Image
import os
import cv2
import glob
import numpy as np
image_list = []
i=01
f1=open("circle-camera.csv",'a')
for filename in glob.glob('extract/*.jpg'):

    hand=cv2.imread(filename)
    hand=cv2.resize(hand,(400,300))
    gray = cv2.cvtColor(hand, cv2.COLOR_BGR2GRAY)
    gray = cv2.GaussianBlur(gray, (5, 5), 0)
    ret, threshold = cv2.threshold(gray,70,255,cv2.THRESH_BINARY)
    im,contours,hierarchy=cv2.findContours
    (threshold,
    cv2.RETR_EXTERNAL,cv2.CHAIN_APPROX_SIMPLE)

    cnt=sorted(contours,key=cv2.contourArea,
    reverse=True)

    I=cv2.drawContours(hand, cnt[0], contourIdx=-1, color=(0,0,255),
    thickness=5)
```

```
M = cv2.moments(cnt[0])

# calculate x,y coordinate of center
cX = int(M["m10"] / M["m00"])
cY = int(M["m01"] / M["m00"])
#print (str(cX) + '\t' + str(cY))

f1.write(str(cX)+';'+str(cY)+'\n')

# put text and highlight the center
cv2.circle(I, (cX, cY), 5, (255, 255, 255), -1)
cv2.putText(I, "centroid",(cX - 25, cY - 25),
cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255, 255, 255), 2)
#cv2.imshow("output",I)

cv2.imwrite('i'+'.jpg',I)
i=i+1
cv2.waitKey(0)
cv2.destroyAllWindows()
```

```
'''
This piece of code detects
the largest contour of the hand
and finds the convexhull
of the detected contour.
Convexhull is one of the property
/feature of a Contour
'''
import cv2 as cv
import numpy as np

# read an image
hand=cv.imread("Hand.jpg")

#resize an image
hand=cv.resize(hand,(400,300))
#convert to grayscale
gray = cv.cvtColor(hand, cv.COLOR_BGR2GRAY)
gray = cv.GaussianBlur(gray, (5, 5), 0)
# threshold an image
ret,threshold= cv.threshold(gray,70,255,cv.THRESH_BINARY)
# use Canny Edge detection
canny_output = cv.Canny(gray, 400,300)

#find contours
im,contours,hierarchy=cv.findContours
(threshold,cv.RETR_EXTERNAL,
cv.CHAIN_APPROX_SIMPLE)

# Find the convex hull object for each
contour
hull_list = []
```

```
for i in range(len(contours)):
    hull = cv.convexHull(contours[i])
    hull_list.append(hull)
# Draw contours + hull results
drawing = np.zeros((canny_output.shape[0], canny_output.shape[1], 3),
dtype=np.uint8)
print(np.shape(contours))
for i in range(len(contours)):

    cv.drawContours(drawing,contours, i, (0,255,0),3)
    cv.drawContours(drawing,hull_list, i, (0,0,255),3)
# Show in a window
cv.imshow('Contours', drawing)
#cv.imwrite('VEx.jpg',drawing)

cv.waitKey()

#Harris Detector
import cv2
import numpy as np
from matplotlib import pyplot as plt

filename = 'output06.jpg'
img = cv2.imread(filename)
img=cv2.resize(img,(600,600))
gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

gray = np.float32(gray)
dst = cv2.cornerHarris(gray,2,3,0.04)
print len(dst)

#result is dilated for marking the corners, not important
dst = cv2.dilate(dst,None)
```

```
# Threshold for an optimal value, it may vary depending on the image.
img[dst>0.01*dst.max()]=[255,0,0]

#cv2.imshow('dst',img)
plt.axis("off")
plt.imshow(img),plt.show()

#if cv2.waitKey(0) & 0xff == 27:
    #cv2.destroyAllWindows()

# -*- coding: utf-8 -*-
"""
Created on Mon Oct 22 14:34:00 2018

@author: Rashmi
good features to track/Shi-Tomasi features detected in an image.
"""
import numpy as np
import cv2
from matplotlib import pyplot as plt

img = cv2.imread('output01.jpg')
gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

corners = cv2.goodFeaturesToTrack(gray,500,0.1,10)
corners = np.int0(corners)
print corners
print len(corners)

for i in corners:
    x,y = i.ravel()
    cv2.circle(img,(x,y),3,255,-1)

plt.imshow(img),plt.show()
```

```python
# -*- coding: utf-8 -*-
"""
Created on Thu Sep 06 14:50:00 2018

@author: rashmi
SIFT features detected in an image
"""
import cv2
import numpy as np
from matplotlib import pyplot as plt

f1=open("SIFT-features.csv",'a')

img = cv2.imread('output01.jpg')
gray= cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
sift = cv2.xfeatures2d.SIFT_create()
kp = sift.detect(gray,None)
#(x1,y1)=kp.pt
i=0
for x in kp: # looping through keypoints
    value=kp[i].pt #printing x,y location of the keypoint
    f1.write(str(value)+'\n')
    i=i+1
print kp[0].size
print len(kp)
#print points2f
#print y1
img=cv2.drawKeypoints(gray,kp,img)
plt.imshow(img)
plt.show()
#cv2.imwrite('sift_keypoints.jpg',img)
#cv2.waitKey(0)

# This piece of code find contours,
```

```
#maximum contour area and
then finds convex hull and
convexity defects for an
#open hand..
# opencv documentation
#Author: rashmi
import cv2
import numpy as np
img = cv2.imread('hand_01.png')
img_gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
ret, thresh = cv2.threshold(img_gray, 127, 255,0)
contours,hierarchy = cv2.findContours(thresh,2,1)[-2:]
c = max(contours, key=cv2.contourArea)
#print len(contours)
#cnt = contours[0]
cv2.drawContours(img_gray, [c], -1, (0, 255, 255), 2)
hull = cv2.convexHull(c,returnPoints = False)
defects = cv2.convexityDefects(c,hull)
for i in range(defects.shape[0]):
    s,e,f,d = defects[i,0]
    start = tuple(c[s][0])
    end = tuple(c[e][0])
    far = tuple(c[f][0])
    cv2.line(img,start,end,[0,255,0],2)
    cv2.circle(img,far,5,[0,0,255],-1)

cv2.imshow('img',img)
cv2.imwrite('HullDefects.png',img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Information Sheet for the participants

Aim of the study: The study aims to record participant's hand gestures for the detection of six distinct Hand Hygiene (HH) Stages as per World Health Organisation's guidelines

Academic Supervisors: Dr Graham Gavin, Dr Jane Courtney, Dr Damon Berry

PhD Researcher: Rashmi Bakshi

Contact: rashmi.bakshi@dit.ie

Standard procedure: Diverse users will be video recorded with the help of ELP-2 Megapixel USB camera, performing HH stages in order to create a rich labelled dataset for gesture classification. Single-handed and two-handed gestures in hand washing will be recorded. Special care will be taken into account to blur the facial expression from the recorded dataset to respect the anonymity of the user in case it occurs. Mostly arm and hand movements of the users will be recorded

Sample size= 20-50 users

<u>Indications for Hand Hygiene activity as per WHO:</u>

- Bare the wrists(short sleeved top or rolled up sleeves)
- Remove all wrist jewellery, including the wrist watch
- Remove all hand jewellery
- Keep fingernails short(tips less than 0.5 cm)
- Do not wear false nails/nail enhancements
- Do not wear nail varnish
- Cover cuts and abrasions with a waterproof dressing

*These guidelines will be handed over to the participants in the form of an informational poster.

Expected duration of the experiment with each participant= 30 minutes

Risk and Benefits: Participating in the study does not imply any risk, however the participants benefit from privileged access to the results and discussion in relation to the awareness campaign regarding Hand Hygiene.

Confidentiality: The survey delivered to the participants is completely anonymous and the only information concerning age, gender and profession are stored digitally. All the papers collected will be destroyed after digitalisation. This information can only be accessed by the scientific supervisors involved in the study. The anonymised results can be relayed to other researchers as well as disseminated.

Dropout: Any participant can drop out of the study at any time. This will have neither advantages nor disadvantages for the participant. A participant can be exempted from the study if he/she does not follow the instructions of the academic supervisors.

The following section is completed by the participant	Yes	No
Have you been fully informed of the nature of this study by the researcher?	☐	☐
Have you had an opportunity to ask questions about this research?	☐	☐
Have you received satisfactory answers to all of your questions?	☐	☐
Have you been full informed of your ability to withdraw participation and/or data from the research?	☐	☐
Have you been fully informed of what will happen to data generated by your participation in the study and how it will be kept safe?	☐	☐
Do you agree to take part in this study, the results of which may be disseminated in scientific publications, books or conference proceedings?	☐	☐
Have your been informed that this consent form shall be kept securely and in confidence by the researcher?	☐	☐
Participant’s Signature		Date:
Researcher’s Signature		Date:

Gender: M/ F ____

Country of Origin:__________

Age:________

Profession:________