Using commercial gesture recognition technology to recognise the South African Sign Language Alphabet: Project Proposal

Anna Borysova
University of Cape Town, Computer Science Department
annasborysova@gmail.com

Shaheel Kooverjee
University of Cape Town, Computer Science Department
skooverjee@gmail.com

Erin Versfeld
University of Cape Town, Computer Science Department
erinversfeld@gmail.com

CSS Concepts

•**Computing methodologies→Machine learning**;
•**Human-centred computing→ Gestural input;**

General Terms

Algorithms, Measurement, Performance, Reliability, Experimentation, Human Factors, Theory, Verification.

Keywords

Machine learning; gesture recognition; South African Sign Language; ICT4D

1. PROJECT DESCRIPTION

There exists a communication barrier between the Deaf and hearing communities globally. Historically, this difference has produced some discord between the linguistic communities, resulting in the Deaf experiencing a reduced set of social opportunities [44]. Throughout this paper the term Deaf as opposed to deaf is used. Deaf refers to a linguistic community whose L1 is a sign language, as opposed to deaf, which refers to a community defined by a medical condition. This project aims to explore the mechanics which will support the building a tool for teaching South African Sign Language (SASL) to the hearing in South Africa. In particular, this study will examine the performance of three gesture recognition devices in conjunction with various machine learning (ML) techniques in identifying gestures from the SASL alphabet. The study's results will inform future development of a tool for teaching SASL. This therefore serves as an investigative project to guide potential future development of a fully-fledged SASL teaching system.

By reducing the communication barrier, addressing the difference in social opportunities afforded to the Deaf and the hearing will hopefully become easier. The Deaf are frequently perceived as being disabled, in part because many hearing in South Africa cannot understand SASL and many Deaf cannot communicate through writing. By providing a means for the hearing to learn SASL affordably, this project hopes to elevate the communication between the groups.

In South Africa's education system there are two primary problems contributing to this barrier. Firstly, the use of Total Communication in most Deaf classrooms [28], a policy in which both SASL and spoken languages are used together. Secondly, the hearing's lack of access to SASL learning material. This latter issue results from SASL not being recognised as a subject for hearing students in South Africa's primary and secondary education system [28]. Learning SASL is costly, as SASL teachers are understandably expensive. The study therefore seeks to address this issue by exploring methods for a system that recognises the SASL fingerspelling gestures. It can only begin to do so as sign languages make use of the entire body to communicate. For example, much meaning is carried in the facial expressions when signing. At this point in time, gesture recognition cannot address all the relevant elements in a simple, easy-to-use, commercially available system. However, the SASL alphabet is gestured with one hand and a system for recognising these gestures could be used to form the beginnings of a tool for teaching SASL to the hearing. The alphabet is often used to sign the names of people and places which do not yet have signs and is often used to introduce hearing learners to the language. The set of gestures in this alphabet can be seen in Figure 1 below.

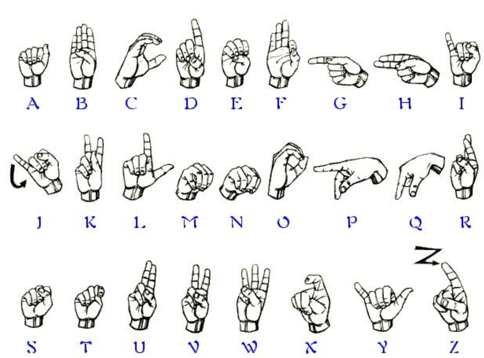


Figure 1. The gestures from the South African Sign Language Alphabet.[1]

[1] https://s-media-cache-ak0.pinimg.com/736x/78/09/f0/7809f0b96480d5a5b74b24ad6c772d16.jpg

2. PROBLEM STATEMENT

Commercially available gesture recognition devices offer a possible means by which SASL can be taught to the hearing. This project aims to investigate various ML techniques' effectiveness for developing such a tool with three different devices: the Microsoft Kinect (Kinect), the Leap Motion Controller (LMC) and the Myo.

2.1 Devices

The Kinect and LMC both implement depth sensing technology, which has been widely applied in gesture recognition. This technology's application to this problem will contribute to the literature surrounding gesture recognition for sign languages. In comparison, the Myo makes use of electromyographic (EMG) sensors in combination with a gyroscope and accelerometer. It is a relatively new and unexplored in comparison to the Kinect and the LMC. Its inclusion will offer fresh insights into gesture recognition for sign languages and gesture recognition as a whole. These devices are commercially available, hence any tool resulting from this study will be usable by anyone owning them. Replication in future studies is also easier, as specifications for custom-built devices do not have to be met.

2.2 Research questions

The primary research question is therefore: *for a given commercially available gesture recognition device, which of the machine learning techniques explored is best for implementing a SASL alphabet learning tool?* The time spent answering this question will be referred to as Phase 1.

Due to the limited duration of this study, the scope has been restricted to look at the three best performing algorithms per device from the literature (Section 5.1 - 3). A fourth algorithm will also be explored. This algorithm will combine the explored algorithms into a single classifier using Bayesian inference methods.

Should time allow, a secondary research question can also be explored. This question would look at how a combination of any of these devices might improve the recognition of these gestures. In other words, the secondary research question is: *for a given combination of the three commercially available gesture recognition devices studied, which of the machine learning techniques explored is best for implementing a SASL alphabet learning tool?* The literature supporting the pursuit of this question is described in Section 5.4. Answering this question will take place in Phase 2, as described further in Section 7.2.2.

2.3 Algorithm evaluation

Answers to these questions will indicate the extent to which SASL learning tools can be developed with available technology. The study's success would mean that it would be worth continuing to explore combinations of other technologies so as to build a tool for recognising more complex SASL gestures. Since the long term vision for where this research would be applied is in an instructional environment, the study must be clear on what is meant by "best for implementing a SASL alphabet learning tool".

Such a tool would need to emphasise correctness. In other words, maximise the number of true positives (TP) and negatives (TN) it produces when classifying gestures, and minimise the number of false positives (FP) and negatives (FN). However, achieving such optimal accuracy is extremely hard, and with that in mind, the classifier with the lowest rate of FPs would be considered the best. Other characteristics against which they will be compared to each other are described in section 3.5.

3. PROCEDURES AND METHODS

Overall, the variety of procedures and methods in the literature seem to indicate that there is no clear standard research methodology to employ in machine learning studies. Therefore the following set of procedures and methods are proposed as they comply with the broader standards set in scientific research.

3.1 Data gathering

There are no existing datasets for recordings of SASL alphabet gestures performed while using one of the target devices. As such, a dataset will have to be created before any algorithms can be implemented.

3.1.1 Stages

The data gathering will consist of three stages: pre-pilot, pilot and test data gathering. The first, the 'pre-pilot' stage, will involve members recording data for themselves, by themselves, to determine the best recording conditions for their individual devices. The subsequent pilot stage will involve designing an environmental setup which satisfies all devices and in which all the devices can be used to record gestures simultaneously. Finally, a larger set of data will be gathered using the same method as in the pilot stage but with participants.

3.1.2 Participants

Up to 50 participants will be recruited from the University of Cape Town student body to perform the SASL alphabet. The participants should all be reasonably familiar with the SASL alphabet. Should fewer than 20 participants be recruited within a week, the study will have to explore alternate avenues of recruitment. Such avenues include, but are not limited to recruiting participants from groups affiliated with the university.

3.1.3 Procedure

Participants will be required to perform five instances of each gesture. These gestures should be performed in a random order. For example, instead of five instances of the gesture for A, five for B, five for C, a participant will be asked to perform the gestures A, Q, F, W, B, V, S, A, until five performances of each is recorded. The set-up for recording these performances is illustrated in Figure 2.

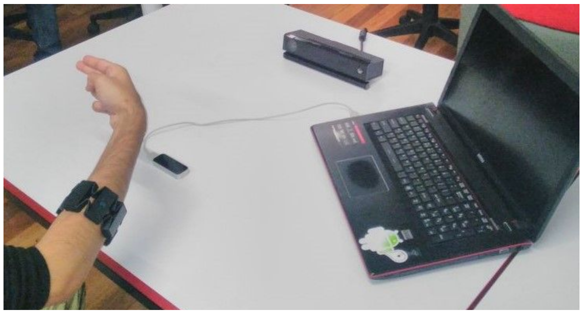


Figure 2. Simultaneous use of devices for data gathering

Fingerspelling, when performed fluently such as by a Deaf signer, is incredibly rapid. As the literature is unclear as to the success of these devices at distinguishing gestures performed in rapid succession, the focus of this study is on individual gestures performed distinctly. Future research can examine more fluid performances.

Data for all devices must be recorded simultaneously, as this allows for the data sets to be used across both phases. This is because two distinctly performed gestures do not produce the same readings even if they are the same gesture. In other words, two performances of the SASL gesture for 'A' by the same person will produce two different readings, even when using the same device both times. Phase 2 requires that data from multiple devices be used to train a classifier to recognise that gesture. This, in turn, requires that this data refers to the same performance of the gesture. Therefore all the devices will need to be employed simultaneously to generate the final training set.

3.1.4 Dataset

The data recorded will consist of individual gestures from the SASL alphabet, rather than sequences. The Kinect and LMC produce images, whilst the Myo produces EMG, accelerometer and gyroscope readings.

This final training set's characteristics are particularly important. A poor training set is detrimental, as documented in the urban legend of an artificial neural network (ANN) designed to identify camouflaged tanks in photographs, but which instead learnt to identify the weather [19]. In order to avoid making this same mistake, the data set for this study must be balanced, and should contain both correct and incorrect gestures, with both major and minor deviations from the form.

3.2 Data processing

As the data will be captured under real-world conditions, it will inevitably require some cleaning and processing prior to being used to train the classifiers. Real-world conditions mean that variables such as muscle density of participants arms, room temperature and lighting conditions will not be controlled. The exact relationship between methods of segmentation, feature extraction and classification is unclear in the literature, therefore several implementations will have to be explored in order to find the correct combination.

3.3 Features

The input to the various classifiers studied is just as important as the classifiers themselves. For this reason feature extraction and feature selection must be carefully considered when setting up and evaluating an ML system.

The Kinect provides a depth image of the camera's view, and extracting the hand region of the image is necessary before any processing can be done. After that, further processing is required to determine the features that will be used by the classifiers. Feature extraction is not of concern with neither the LMC nor the Myo. In both cases, feature selection is used instead.

3.4 Gesture Classifiers

Once features have been obtained from the gathered data, these will then be fed into various classifiers, in order to learn recognition of the gestures. The following classifiers have been selected for investigation in this study based on their performance in previous work. These will be trained on the dataset collected (see Section 3.1), a portion of which will be reserved for evaluating the classifiers (see Section 3.5).

3.4.1 Support Vector Machines

A support vector machine (SVM) is a supervised ML technique that is mostly used for classification problems. After plotting the training data points in an n-dimensional space, a hyperplane with maximal margin width is found to differentiate the data. New data instances can then be classified according to the optimal boundaries formed.

For this project, multi-class SVMs will be used to classify the data into more than two classes, as opposed to basic binary SVM classifiers.

3.4.2 Hidden Markov Models

Hidden Markov Models (HMMs) define a Markov model on a set of hidden or latent variables. The observed or visible variables are dependent upon these, forming a joint distribution [5, 8, 16].

One of the more powerful properties of HMMs is their ability to exhibit some degree of invariance to local time warping. In speech recognition the warping of the time axis is associated with the natural variations in the speed of speech [7]. A similar warping can be expected in this study, as participants perform gestures at differing speeds. HMMs are able to accommodate this variation and hence are well suited to this domain.

3.4.3 *Artificial Neural Networks*

An ANN consists of layers of nodes interconnected by 'neurons' of varying weights [18]. Information is accepted at the input layer of nodes, processed through some number of hidden layers, and finally, the output layer of nodes outputs the result.

ANNs' ability to learn through reinforcement means that they are able to generalise and associate data. With successful training an ANN can find reasonable solutions for similar problems of the same class, even if it is not explicitly trained to address them. This in turn means that ANNs have a high fault tolerance for noisy data [25]. This property is useful in this study's context, as there are many variables which affect the detection of gestures when performed by different people.

3.4.4 *K-nearest neighbour*

The K-nearest neighbour algorithm improves on the nearest neighbour algorithm by drawing on more neighbours. In so doing it produces a more robust classifier and a smoother decision boundary. When K is very large, the classifications tend towards being the same. Therefore cross validation is utilised to obtain the most optimal value for K [5].

K-nearest neighbour has performed well in studies which utilised a combination of static and dynamic gestures. Given that such a mixture will be explored in this study, it will be useful to include this classifier.

3.4.5 *Naïve Bayes Classifier*

This classifier models the probability of the class variable using the simplifying assumption that each feature in the feature vector is independent [9, 40]. The probabilistic model developed for generating the predictions is derived from Bayes theorem.

$$P(C = \lambda_j \mid x_i = y) = \frac{P(xi = y \mid C{=}\lambda j)P(C{=}\lambda j)}{P(xi = y)} \qquad (3.1)$$

This model can be interpreted as the probability that the class C will have the given value $_j$ when given the feature vector x has the values y. After expanding this equation to explicitly include each individual feature and then applying the chain rule, a naïve assumption is made. Here it is assumed that $P(y_i|\lambda_j, y_i) = P(y_i|\lambda_j)$. The classifier is then built using the resultant equation and a decision rule.

3.5 Evaluation

Each of the classifiers used for a given device can be compared against one another using quantitative measures. These results can also be compared against those of previous studies.

There are several standard quantitative measures of the performance of ML techniques. The first of these is the accuracy or recognition rate. This refers to the percentage of test set tuples which are correctly identified by the classifier. The rate at which a classifier produces type I and II errors is another common measure, and is typically explored using a confusion matrix. Other measures of a classifier's accuracy include its model accuracy (3.5.1.1), misclassification rate (3.5.1.2), sensitivity (3.5.1.3) and specificity (3.5.1.4).

$$\frac{TP + TN}{TP + FP + TN + FN} \qquad 3.5.1.1$$

$$\frac{FP + FN}{TP + FP + TN + FN} \qquad 3.5.1.2$$

$$\frac{TP}{TP + FN} \qquad 3.5.1.3$$

$$\frac{TN}{TN + FP} \qquad 3.5.1.4$$

Receiver Operating Characteristic (ROC) curves are another useful quantitative measure of accuracy. These graphs depict how the relationship between true and false positives change. The greater the area under this curve, the more accurate the model. ROC curves are ideal for visual comparison of models. Cross validation can then be used to evaluate the final model.

A popular cross validation method is the K-fold method. In this evaluation, the data set is divided into K folds. Then for each of K experiments, the Kth fold is reserved for testing whilst the rest is utilised for training. The system's accuracy is then calculated as the average error rate across K folds.

4. ETHICAL, PROFESSIONAL AND LEGAL ISSUES

As all the members of this team, supervisors included, are hearing, there is an ethical concern that this tool is being designed for the benefit of the already socially dominant hearing, rather than for the Deaf, and that this reinforces the social divide between the two in favour of the hearing. However, the team sees it instead as being a solution to the side of the problem with which they are familiar, and one which forces the dominant group to take responsibility to address the issue, rather than forcing the marginalised group to address it. Nevertheless, as the project unfolds the team will consult with members of the Deaf community and members of staff with experience doing similar research to ensure that the study remains sensitive to these issues.

The University of Cape Town (UCT) holds copyright on all intellectual material produced by its postgraduate students, and promotes publishing under a Creative Commons license. This should be borne in mind when applying for ethical clearance from the Faculty of Science Research Ethics Committee.

The ethical clearance will be required to recruit participants for the gathering of training data. This will require participants to perform the SASL alphabet gestures several times with the use of the devices. This data will be used to train the classifiers described in Section 3.4. All participants shall be recruited and have their data treated in accordance with standard ethical practices, such as informed consent and anonymisation.

5. RELATED WORK

Gesture recognition as a whole is a well-researched topic, and a significant number of studies have already applied it to sign languages. This study extends this already existing research and applies it to SASL.

5.1 Kinect

Previous studies have demonstrated the strengths and weaknesses of using the Kinect in gesture recognition studies [2, 11, 21, 39, 51]. Its strengths include its affordability, reliable depth map obtained from the depth sensor [51], not requiring background image calibration or markings or gloves for tracking [21] and robustness across environmental settings [2, 21]. However, the camera's low resolution makes it challenging to find and separate specific objects in the image [11]. For this study, where hands play such an important role, this shortcoming makes interpreting gestures challenging, as the hands are both the most important feature for recognition of fingerspelling gestures and occupy a small portion of the image.

As far as methods for classifying data from the Kinect are concerned, support vector machines (SVMs) [2, 11, 17], artificial neural networks (ANNs) [4, 36] and hidden Markov models (HMMs) [41, 42, 50] have all been used with some success in the past. Other methods explored include dynamic time warping (DTW) [21, 41], histogram of oriented gradient (HOG) [21, 49] and HAGR-D [41]. SVMs, ANNs and HMMs are better researched and understood on the whole, and hence will be utilised in this study. Each require upwards of 100 samples per gesture for adequate training prior to testing. Once these models have been trained they will be tested in accordance with the evaluation methods described in Section 3.

5.2 Leap Motion Controller

The LMC is similar to the Kinect in that it is also a depth-based gesture recognition tool. Unlike the Kinect, however, there is no accessible depth map [38]. Previous versions of its firmware had issues with recognising gestures accurately when some fingers are obscured or too close together [38]. Unfortunately, this cannot be resolved by merely using more LMCs to record the gesture from all angles [34]. It is unclear whether or not this issue has been resolved by updated firmware, but will be an issue for this study if it persists.

Studies with the LMC have had similar success to those with the Kinect when using SVMs [31, 32, 48] and ANNs [30, 33]. Other successful ML techniques have been k-nearest neighbour [14, 47], random forests [10, 14], Naïve Bayes Classifier [33], Dempster-Shafer [34], linear discriminant analysis (LDA) [34], and HMMs [12]. As SVMs and ANNs are already being investigated within the context of the Kinect, it will be useful to consider them for the LMC too, so as to be able to draw useful inferences from a comparison of the relative performance. The third classifier tested will be the Naïve Bayes Classifier (NBC) to explore a wider variety of potential classifiers.

5.3 Myo

The Myo differs from the Kinect and LMC in that it performs electromyographic (EMG) based gesture recognition rather than depth. It includes inertial measurement unit (IMU) sensors in addition to its EMG sensors, a configuration which has been confirmed to have worked well in similar hardware [13, 14, 20, 26]. It comes in the form of an armband, which is then positioned on the forearm. One of the shortcomings of this device is that, unlike the custom hardware used in previous studies [13, 20, 24, 35, 51], the muscles from which data is received when a gesture is performed cannot be specified. The literature indicates that this could be why it has a low rate for recognising fine gestures [1]. However, [29, 45, 46] all indicate that this shortcoming can be overcome with the correct ML technique.

Unlike the Kinect and the LMC, SVMs do not appear to work well with the Myo [1, 35]. However, similar to these two devices, ANNs [3, 29] and HMMs [20, 26, 27] can be used to accurately recognise gestures using the Myo. Much like the LMC, LDA [23, 26] and a form of Bayesian Classifier (specifically the Bayesian Linear Classifier) [13, 27] can be employed to recognise gestures with high recognition rates. Finally, k-nearest neighbour [47] produces similarly positive results. To maintain some homogeneity, ANNs and HMMs should also be evaluated here. K-nearest neighbour will also be employed due to its ability to handle both static and dynamic gestures.

5.4 Combining devices

Research suggests that gesture recognition can be enhanced by combining two or more distinct devices [15, 31, 32, 37]. Previous research indicates that this advantage will not be gained by utilising two of the same type of device [34], hence we will explore combinations of different devices.

The combination of the LMC and Kinect has been particularly well explored, with multi-class SVMs and fusing data via the corresponding point set registration algorithm [6] appearing to work particularly well [31, 37]. Applying these methods and others, such as Bayesian inference classifiers [52], to SASL will indicate the generalisability of these findings across sign languages.

The use of Myo and Kinect to recognise sign language gestures appears not to have received very much attention, although at least one study has explored this combination in the context of medicine [53]. After a data level fusion of the readings from both devices, a combination of multi-dimensional dynamic time warping and long short-term memory was successfully used to implement a system with both devices. The combination of LMC and Myo is similarly poorly researched. A study suggested that fusing the data from both devices using a Kalman filter may be a good way to go about combining the devices [43].

6. ANTICIPATED OUTCOMES

The study will contribute to the academic communities understanding of the applicability of machine learning and various gesture recognition technologies to recognising rudimentary sign language gestures. It will lay the groundwork for future research into the recognition of more complex sign language gestures.

7. PROJECT PLAN AND WORK ALLOCATIONS

7.1 Risks

Every project has associated risks. In Table 1, the risks for this project have been outlined and are presented alongside their corresponding mitigation, monitoring and management strategies. The probability of these risks occurring and their impact on the project are measured on a scale of 1 - 10, with 1 being low and 10 high.

7.2 Timeline and milestones

The project's proposed timeline is shown in Figure 1 in the Appendix. This timeline covers all deliverables from the proposal demonstration until the final reflection paper

Table 2 provides a summary of the major milestones. Broadly, the project can be split into Phase 1 and Phase 2, which Phase 2 being a completely separable extension.

7.2.1 Phase 1

Phase 1 involves preliminary data gathering and implementation of the algorithms decided on in Section 5 for each of the devices. The preliminary data gathering consists of a pilot phase and a pre-pilot phase to ensure smooth functioning of the final data gathering stage. The pilot will leave usable data to train and test the algorithms, making Phase 1 a viable project in its own right. The algorithms which will be tested in Phase 1 are as follows:

1. **Kinect:** ANN, SVM, and HMM
2. **LMC:** ANN, SVM, and NBC
3. **Myo**: ANN, HMM, and k-nearest neighbour

A fourth method which combines the best performing classifiers using bayesian inference will also be explored for each device.

7.2.2 Phase 2

Phase 2 will move into experimentation of various combinations of the target devices and their algorithms. This phase will contain the final data collection, which will provide a wider range of data with which to train and test the system.

Table 2. Proposed project milestones

Date*	Milestone
14 June	Proposal presentation
21 June	Ethical clearance application submitted
30 June	Proposal finalised and uploaded to Vula
	End Phase 1
18 August	Software feasibility demo
28 August	End Phase 2
5 September	First drafts of final papers
12 September	Final drafts of final papers
22 September	Final paper submissions
2 October	Final code submissions
9 October	Final project demonstrations
	Poster due
12 October	Web page due
23 October	Reflection paper due

*All dates taken to be in the year 2017

7.3 Resources required

One of each of the devices (Kinect, LMC and Myo) is required, and will be provided by UCT Computer Science Department and the team members. The team members will need to construct training data for each of these devices. Open source libraries will be used to clean the data and facilitate the ML techniques.

7.4 Deliverables

Each team member will deliver a system that answers their respective research questions. The training data sets produced will form part of this deliverable, as well as papers documenting their individual research. This research should be reproducible and contribute to the broader scientific knowledge base on this topic.

7.6 Work allocation

Each team member will be required to clean and segment the data set for their respective device. In addition to this, each member will be responsible for exploring one of the devices in answering the primary research question, as follows:

1. **Anna Borysova:** LMC
2. **Shaheel Kooverjee:** Kinect
3. **Erin Versfeld:** Myo

Concern has been raised that the Myo make be a heavier workload than that of the LMC or Kinect. This is largely attributed to the relatively small body of literature surrounding the device compared to the other devices. However, this difference is due to the fact that the Myo is relatively new. The literature does not indicate that any of the devices is more difficult to work with, hence the current work split is deemed to be fair.

Whilst the Myo may also appear to be the more interesting of the devices to explore given how new it is to the literature, the other devices also offer new insights into this body of knowledge. They have never been applied to recognising SASL gestures, and the ground work in Phase 1 allows for exploration of combinations of devices in Phase 2.

Should there be time for Phase 2, researchers will examine their results from Phase 1 and use them to determine how best to combine the data from these devices. The work will then be split up appropriately.

8. REFERENCES

[1] J. G. Abreu, J. M. Teixeira, L. S. Figueiredo. and V. Teichrieb. Evaluating sign language recognition using the Myo armband. in *2010 XVII Symposium on Virtual and Augemented Reality*, (2016), IEEE, 64-70

[2] Anant Agarwal and Manish K akur. 2013. Sign Language Recognition using Microso Kinect. In Sixth International Conference on Contemporary Computing (IC3). IEEE, Noida, India, 181–185.

[3] V. Amatanon, S. Chanhang, P. Naiyanetr, and S. Thongpang. Sign language-Thai alphabet conversion based on electromyogram (EMG). in *2014 7th Biomedical Engineering International Conference*, (2014), IEEE, 1-4

[4] Jos Barata, Joo Delgado, and Tiago Cardoso. 2015. Hand Gesture Recognition towards Enhancing Accessibility. *Procedia Computer Science* (2015), 419–429.

[5] David Barber. 2012. *Bayesian reasoning and machine learning*. Cambridge University Press.

[6] Paul J. Besl and Neil D. McKay. 1992. A Method for Registration of 3-D Shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence - Special issue on interpretation of 3-D scenes - part II (1992), 239–256.

[7] Christopher M. Bishop. *Pattern recognition and machine learning.* Springer.

[8] B. E. Boser, I. M. Guyon, and V. N. Vapnik. 1992. A Training Algorithm for Optimal Margin Classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, (1992), 144–152.

[9] Pascal Brandt. 2014. *An investigation of multi-label classification techniques for predicting HIV drug resistance in resource limited settings.* Master's thesis. University of Kwa-Zulu Natal, Durban, South Africa.

[10] Leo Breiman. 2001. Random forests. *Machine learning* 45, 1 (2001), 5.

[11] Chao Sun, Tianzhu Zhang, and Changsheng Xu. 2015. Latent Support Vector Machine Modeling for Sign Language Recognition with Kinect. *ACM Transactions on Intelligent Systems and Technology (TIST)* (2015), Special Section on Visual Understanding with RGB–D Sensors.

[12] Chen Yanmei, Ding Zeyu, Chen Yen-Lun and Wu Xinyu. 2015. Rapid recognition of dynamic hand gestures using leap motion. In *Information and automation, 2015, IEEE International Conference on*, IEEE, (2015), 1419 - 1424.

[13] X. Chen, X. Zhang, Z. Y. Zhao, J. H. Yang, V. Lantz, and K. Q. Wang. Hand gesture recognition research based on surface EMG sensors and 2D-accelerometers. in *2007 11th IEEE International Symposium on Wearable Computers*, (2007), 11-14

[14] Hong Cheng, Lu Yang, and Zicheng Liu. 2016. Survey on 3D Hand gesture recognition. *IEEE Transactions on Circuits and Systems for Video Technology* 26, 9 (2016), 1659 - 1673.

[15] Andrew Clark and Deshendran Moodley. 2016. A system for a hand gesture-manipulated virtual reality environment. In *Proceedings of the Annual Conference of the South African Institute of Computer Scientists and Information Technologists.* ACM, 10.

[16] C. Cortes and V. Vapnik. 1995. Support-Vector Networks. *Machine Learning,* 20 (1995), 273–297.

[17] Navneet Dalal and Bill Triggs. 2005. Histograms of Oriented Gradients for Human Detection. In *International Conference on Computer Vision & Pattern Recognition*. IEEE, San Diego, United States, 886–893.

[18] Walter H Delashmit and Michael T Manry. 2005. Recent developments in multilayer perceptron neural networks. In Proceedings of the seventh Annual Memphis Area Engineering and Science Conference, MAESC. Citeseer.

[19] Neil Fraser. 1998. Neural Network Follies. (September 1998). Retrieved 20 May 2017 from https://neil.fraser.name/writing/tank/

[20] Georgi, M., Amma, C., & Schultz, T. Recognizing hand and finger gestures with IMU based motion and EMG based muscle activity sensing. *BIOSIGNALS*, 99-108

[21] Frank Huang and Sandy Huang. 2011. Interpreting American Sign Language with Kinect. Stanford University term paper for CS 299 (2011).

[22] Pat Jangyodsuk, Christopher Conly, and Vassilis Athitsos. 2014. Sign Language Recognition using Dynamic Time Warping and Hand Shape Distance Based on Histogram of Oriented Gradient Features. In *Proceedings of the 7th International Conference*

on Pervasive Technologies Related to Assistive Environments. ACM, Island of Rhodes, Greece, Article No. 50.

[23] S. Jiang, B. Ly, X., Shang, C. Zhang, H. Wang, and P. B. Shull. Development of a real-time hand gesture recognition wristband based on sEMG and IMU sensing. in *2016 IEEE International Conference on Robotics and Biomimetics*, (2016), 1256-1261

[24] V. E. Kosmidou, L. J. Hadjileontiadis, and S. M. Panas. Evaluation of surface EMG features for the recognition of American Sign Language gestures. in *Engineering in Medicine and Biology Society, 2006. 28th Annual International Conference of the IEEE*, (2006), 6197-6200

[25] David Kriesel. 2007. *A brief introduction to neural networks*.

[26] Y. Li, X. Chen, J. Tian, X. Zhang, K. Wang, and J. Yang. Automatic recognition of sign language sub-words based on portable accelerometer and EMG sensors. in *International Conference on Multimodal Interfaces and the Workshop on Machine Learning for Multimodal Interaction*, (2010), 17

[27] Z. Lu, X. Chen, Q. Li, X. Zhang, P. and Zhou. A hand gesture recognition framework and wearable gesture-based interaction prototype for mobile devices. *IEEE transactions on human-machine systems, 44*(2), 293-299

[28] L. Magongwa. (2010). Deaf education in South Africa. *American Annals of the Deaf, 155*(4), 493-496

[29] A. L. Madushanka, R. G, Senevirathne, L. M. Wijesekara, S. M. Arunatilake, and K. D. Sandaruwan. Framework for Sinhala Sign Language recognition and translation using a wearable armband. in *2016 sixteenth International Conference on Advances in ICT for Emerging Regions*, (2016), 49-57

[30] Rajesh B Mapari and Govind Kharat. 2016. American Statis Signs Recognition Using Leap Motion Sensor. In *Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies*. ACM, 67.

[31] Giulio Marin, Fabio Dominio, and Pietro Zanuttigh. 2014. Hand gesture recognition with leap motion and kinect devices. In *Image Processing (ICIP), 2014 IEEE International Conference on*. IEEE, 960 - 965.

[32] Guilio Marin, Fabio Dominio, and Pietro Zanuttigh. 2016. Hand gesture recognition with jointly calibrated Leap Motion and depth sensor. *Multimedia Tools and Applications*, 75, 22 (2016), 14991-15015.

[33] M. Mohandes, S. Aliyu, and M. Deriche. 2014. Arabic sign language recognition using the leap motion controller. In *Industrial Electrontics (ISIE), 2014 IEEE 23rd International Symposium on*. IEEE, 960 - 965.

[34] M. Mohandes, S. Aliyu, and M. Deriche. 2015. Prototype Arabic Sign Language recognition using multi-sensor data fusion of two leap motion controllers. In *Systems, Signals & Devices (SSD), 2015 12th International Multi-Conference on.* IEEE, 1 - 6.

[35] G. R. Naik, D. K. Kumar, and Jayadeva. Twin SVM for gesture classification using the surface electromyogram. *IEEE Transactions on Information Technology in Biomedicine, 14*(2), 301-308.

[36] Ednaldo Brigante Pizzolato, Mauro dos Santos Anjo, and Sebastian Feuerstack. 2012. A Real-Time System to Recognize Static Gestures of Brazilian Sign Language (Libras) alphabet using Kinect. In *IHC ´12 Proceedings of the 11th Brazilian Symposium on Human Factors in Computing Systems*. SBC, Cuiab, MT, Brazil, 259–268.

[37] Benot Penelle and Olivier Debeir. 2014. Multi-sensor data fusion for hand tracking using Kinect and Leap Motion. In Proceedings of the 2014 Virtual Reality International Conference. ACM, Laval, France, Article No. 22.

[38] Leigh Ellen Potter, Jake Araullo, and Lewis Carter. 2013. The leap motion controller: a view on sign language. In *Proceedings of the 25th Australian computer-human interaction conference: augmentation, application, innovation, collaboration.* ACM, 175-178.

[39] Zhou Ren, Jingjing Meng, Junsong Yuan, and Zhengyou Zhang. 2011. Robust hand gesture recognition with kinect sensor. In *19th ACM international conference on Multimedia*. ACM, Scosdale, Arizona, USA, 759–760.

[40] I. Rish, "An Empirical Study of the Naive Bayes Classifier," Proceedings of the IJCAI Workshop on Empirical Methods in Artificial Intelligence 3, 41–46 (2001).

[41] Diego G. Santos, Bruno J. T. Fernandes, and Byron L. D. Bezerra. 2015. HAGR-D: A Novel Approach for Gesture Recognition with Depth Maps. *Sensors 15*(11) (2015), 28646–28664.

[42] Noha A. Sarhan, Yasser El-Sonbaty, and Sherine M. Youssef. 2015. HMM-based Arabic Sign Language Recognition using Kinect. In *Tenth International Conference on Digital Information Management (ICDIM)*. IEEE, Jeju, 169 – 174.

[43] Eider C. P. Silva, Estaban W. G. Clua, and Anselmo A. Montenegro. 2015. Sensor data fusion for full arm tracking using Myo armband and Leap motion. In *2015 14th Brazilian Symposium on Computer Games and Digital Entertainment*. IEEE, 128 - 134.

[44] L. Simms, M. Rusher, J. F. Andrews, and J. Coryell. Apartheid in Deaf education: Examining workforce diversity. *American Annals of the Deaf, 153*(4), 384-395.

[45] A. Srisuphab, and P. Silapachote. Artificial neural networks for gesture classification with inertial motion sensing armbands. *Region 10 Conference (TENCON), 2016 IEEE*, (2016), 1-5

[46] J. Taylor. Real-time translation of American Sign Language using wearable technology.

[47] Paulo Trigueiros, Fernando Riberio, and Luis Paulo Reis. 2014. Generic system for human-computer gesture interaction. In *Autonomous Robot Systems and Competitions (ICARSC), 2014 IEEE International Conference on*. IEEE, 175 - 180.

[48] Merkourios Simos and Nikolaos Nikolaidis. 2016. Greek sign language alphabet recognition using the leap motion device. In *Proceedings of the 9th Hellenic Conference on Artificial Intelligence*. ACM, 34.

[49] Haijing Wang, Alexandra Stefan, Sajjad Moradi, Vassilis Athitsos, Carol Neidle Neidle, and Farhad Kamangar. 2012. A System for Large Vocabulary Sign Search. In *Proceedings of the 11th European conference on Trends and Topics in Computer Vision Part I, ECCV10*. Springer-Verlag, Berlin, Heidelberg, 342–353.

[50] Zahoor Zafrulla, Helene Brashear, ad Starner, Harley Hamilton, and Peter Presti. 2011. American sign language recognition with the Kinect. In *13th international conference on multimodal interfaces*. ACM, Alicante, Spain, 279–286

[51] X. Zhang, X. Chen, Y. Li, V. Lantz, K. Wang, and J. Yang. A framework for hand gesture recognition based on accelerometer and EMG sensors. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 41*(6), 1064-1076

[52] Pujan Ziaie, Thomas Muller, and Alois Knoll. 2008. A novel approach to hand-gesture recognition in a human-robot dialog system. *Image Processing Theory, Tools and Applications, 2008. IPTA 2008. First Workshops on*. IEEE.

[53] Tian Zhou and Juan P. Wachs. 2016. Early turn-taking prediction in the operating room. In *2016 AAAI Fall Symposium Series: Artificial Intelligence for Human-Robot Interaction*. Association for the Advancement of Artificial Intelligence, 117 - 123.

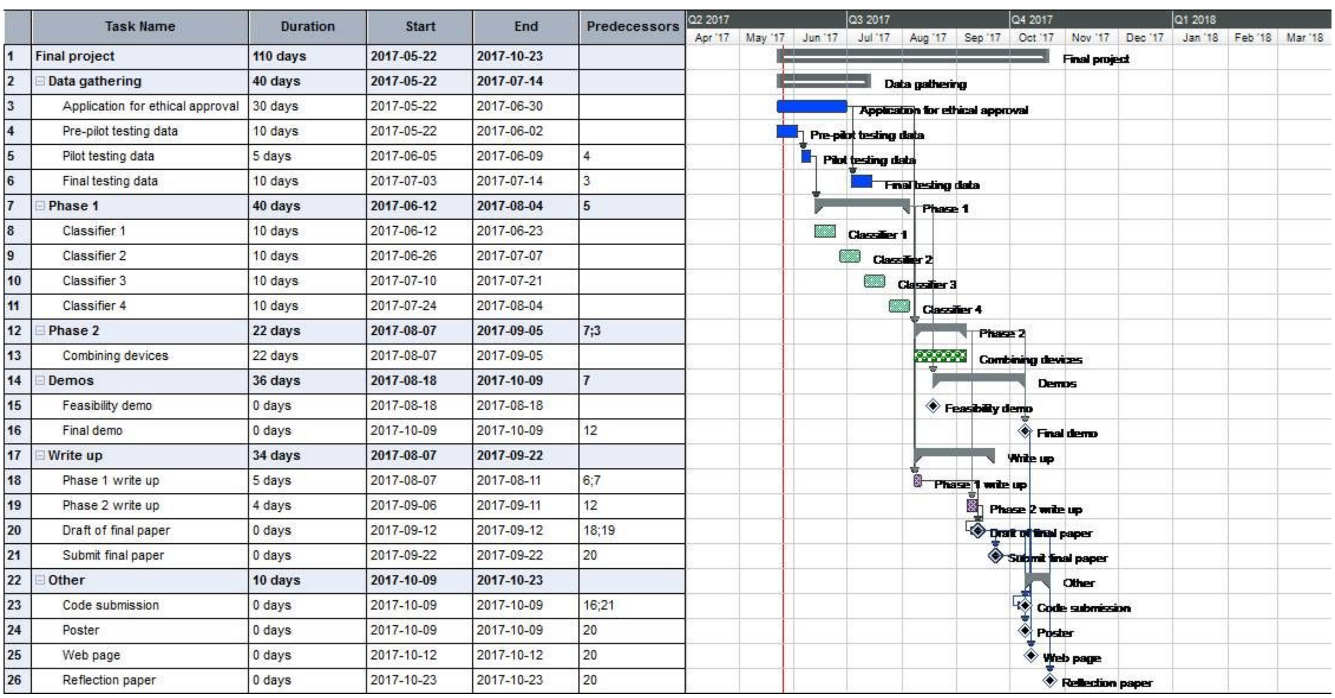
9. APPENDICES

Table 1. Risk matrix including the mitigation, monitoring and management strategies for each risk

Risk	Prob.	Impact	Consequence	Mitigation	Monitoring	Management
Unable to find or use relevant ML libraries.	2	10	Team unable to build and test proposed system and therefore unable to complete project.	Research available libraries and support available for them.	Obtain advice from members of department who have experience using such libraries.	Implement relevant methods internally, or replace method.
Ethical clearance denied.	5	8	Team unable to perform user testing.	Consult members of department who have done similar research. Carefully consider all ethical concerns prior to submitting application.	Consult Ethics Board to track progress of the application.	Split training data up so that a portion of it is kept aside to test system. Re-apply for clearance.
Insufficient training data.	5	8	Project results are invalid as a technique could perform better with more training.	Record more data than anticipated to be necessary, basing this volume off of previous work.	Ensure that techniques are performing similarly to previous work.	Generate more training data.
Team fails to meet project deadline.	2	10	Team will not be able to hand in the project on time and the project will be considered a failure.	Control the scope of the project, making sure that all the necessary milestones are met.	Ensure team proceeds according to timeline.	Drop Phase 2.
Team member fails to meet end of phase.	3	5	Phase end milestones require team members to be in sync. If one fails to meet these milestones, the project's progress will be delayed.	Facilitate clear and open communication channels to detect issues early and reallocate work as required.	Maintain communication channels between team members.	Drop Phase 2.
Team member discontinues their involvement in project.	1	3	Individual projects are not tightly coupled, therefore project as a whole will not be delayed although its value will be somewhat diminished.	Facilitate clear and open communication channels to detect issues early and reallocate work as required.	Maintain communication channels between team members.	Reallocate work to ensure that the project remains on schedule.
Scope creep occurs.	4	6	Time could be spent on unnecessary features and paths of exploration, leading to delays in meeting milestones	Ensure all necessary features have been implemented and milestones complete before working on additional features.	Maintain high levels of reporting between team members and supervisors.	Discard unnecessary features and refocus on features which satisfy the project's aims and goals.

Equipment failure occurs.	3	10	No data will be able to be recorded, and hence no evaluation of algorithms will be able to take place	Use devices as carefully as possible.	Use devices regularly over data gathering periods to ensure devices still produce expected outputs.	Obtain backup devices.
Insufficient participants are recruited.	4	10	The dataset will be unbalanced potentially resulting in classifiers misclassifying gestures or performing suboptimally.	Approach as many interested student organisations are possible.	Maintain a count of the number of recruited participants and ensure that levels are met on schedule.	Approach interested groups outside of UCT but which are affiliated with the university.
Unbalanced dataset is generated.	7	10	Classifiers will potentially misclassify gestures or perform suboptimally.	Recruit diverse participants.	Keep track of the quality of all of the performances recorded and ensure that there is adequate variety.	Recruit more participants.

Figure 1. A Gantt chart diagramming the proposed timeline for the project and its associated tasks



	Task Name	Duration	Start	End	Predecessors
1	Final project	110 days	2017-05-22	2017-10-23	
2	Data gathering	40 days	2017-05-22	2017-07-14	
3	Application for ethical approval	30 days	2017-05-22	2017-06-30	
4	Pre-pilot testing data	10 days	2017-05-22	2017-06-02	
5	Pilot testing data	5 days	2017-06-05	2017-06-09	4
6	Final testing data	10 days	2017-07-03	2017-07-14	3
7	Phase 1	40 days	2017-06-12	2017-08-04	5
8	Classifier 1	10 days	2017-06-12	2017-06-23	
9	Classifier 2	10 days	2017-06-26	2017-07-07	
10	Classifier 3	10 days	2017-07-10	2017-07-21	
11	Classifier 4	10 days	2017-07-24	2017-08-04	
12	Phase 2	22 days	2017-08-07	2017-09-05	7;3
13	Combining devices	22 days	2017-08-07	2017-09-05	
14	Demos	36 days	2017-08-18	2017-10-09	7
15	Feasibility demo	0 days	2017-08-18	2017-08-18	
16	Final demo	0 days	2017-10-09	2017-10-09	12
17	Write up	34 days	2017-08-07	2017-09-22	
18	Phase 1 write up	5 days	2017-08-07	2017-08-11	6;7
19	Phase 2 write up	4 days	2017-09-06	2017-09-11	12
20	Draft of final paper	0 days	2017-09-12	2017-09-12	18;19
21	Submit final paper	0 days	2017-09-22	2017-09-22	20
22	Other	10 days	2017-10-09	2017-10-23	
23	Code submission	0 days	2017-10-09	2017-10-09	16;21
24	Poster	0 days	2017-10-09	2017-10-09	20
25	Web page	0 days	2017-10-12	2017-10-12	20
26	Reflection paper	0 days	2017-10-23	2017-10-23	20