MAST: Myo Armband Sign-Language Translator for Human Hand Activity Classification

Zuhaib Muhammad Shakeel*, Soonhyuk So*, Patrick Lingga†, and Jaehoon (Paul) Jeong*
* Department of Computer Science & Engineering, Sungkyunkwan University, Suwon, Republic of Korea
† Department of Electrical & Computer Engineering, Sungkyunkwan University, Suwon, Republic of Korea
Email: {goldshakil, synops1s, patricklink, pauljeong}@skku.edu

***Abstract*—As Computer Science has grown into an encompassed field in various scientific areas, the need for developing a computer aided and artificially intelligent device has become more important especially in the medical field. Artificial Intelligence (AI) plays a vital role not only in accelerating and optimizing common tasks but also in performing tasks that humans are incapable of. This paper presents a Myo Armband Sign-Language Translator (MAST), which is a novel algorithm to translate a hand's gestures into medical sign language using a Myo armband sensor which collects muscles' electromyography signals and then to classify them using an enhanced version of a dynamic random forest. Our experimental results indicate that a systematic fine tuning of MAST parameters leads to an accuracy improvement of 13% over the state-of-the-art scheme such as SCIKIT's random forest. Other comparison results show an improvement of over 20% compared to a popular classification scheme such as Support Vector Machines (SVM) and a deep learning technique such as Convolutional Neural Network (CNN).**

***Index Terms*—Electromyography, Myo Armband, Sign-language Translator, Random Forest, Support Vector Machine, Convolutional Neural Network**

I. Introduction

Over the course of the last couple of years, there has been a significant increase in the abundant input and interaction devices with computers. The core role of such devices is to bring a technology closer to the reality and to perform hard or even seemingly impossible tasks in the past. Moreover, there has been ongoing research in gesture recognition for hearing-impaired and mute people which takes advantage of such input gadgets to attain high accuracy while maximizing the convenience for the users [1].

Since mute individuals have communication problems while dealing with other people and they basically rely on gestures to communicate every day, it is essential to classify hand gestures into written text so that the communication becomes comprehensible. To carry out such classification, the fundamental step is to get raw data from hand gestures and feed it into computer systems and this data can be gathered from data gloves, vision and image-based system, and electromyogram (EMG) sensors [2]. Electromyography is a technique for assessing and recording the electrical activity produced by skeletal muscle contractions. That is, EMG signals show the activity level of a specific muscle and these bio-electric signals can be picked up using sensors attached to the body like Myo armband [3].

This paper proposes a Myo Armband Sign-Language Translator (MAST) that is a classifier which classifies EMG signals corresponding to medical sign language retrieved from Myo armband into written text. The EMG raw data can be collected, featured using linear discriminant analysis, and finally used as an input for a novel dynamic random forest classifier for our MAST. The Random Forest (RF) algorithms form a family of classification methods that rely on the combination of several decision trees such that each decision tree contributes to the final ensemble and voting scheme. Even though there are a lot of deep learning and machine learning techniques that have been used in the past, to the best of our knowledge, this paper is the first work that tackles the recognition of medical sign language using a dynamic random forest. The main contributions of this paper that make our MAST novel when compared to previous schemes are listed as follows:

- An optimized version of a dynamic random forest which outperforms the state-of-the-art such as SCIKIT's implementation and also outperforms a machine learning (i.e., SVM) and a deep learning technique (i.e., CNN);
- A linear discriminant analysis scheme to feature EMG signals;
- A practical and an easily trainable architecture for classifying EMG signals into hand gestures, which can be used in any relevant field.

The remainder of this paper is composed as follows. Section II summarizes the related work of hand gesture classification. Section III describes our MAST architecture, implementation, and key features. In Section IV, the performance evaluation is presented for our MAST and other baselines. Section V concludes this paper along with future work.

II. Related Work

This section describes the related work that has been done in this area in a more systematic way. It is essential to mention that most of the research which has targeted this field is either dependent on Statistical Machine Translation (SMT) or a deep learning approach. Thus, it is really important to describe those mechanisms before diving into the details of the papers that are based on them. First of all, SMT is regarded as a sub-field of natural language processing that investigates how to automatically translate text or speech across human languages using probabilistic models from parallel corpora, and hence it needs a large volume of training data to build

such probabilistic models [4]. On the other hand, Computer Vision-based systems focus on capturing images and then they classify those images into objects using different CNN models [5].

The work proposed in [6] is one of the most notable works in this area in which the authors propose a custom sign language translation system that uses a specialized glove programmed with an SMT algorithm to translate 20 gestures into 20 English letters with an accuracy of 96%. Although the accuracy of this scheme is high, the practicality of using such a glove is debatable since it is only able to translate limited English letters rather than actual phrases or words, so it is not cost-efficient if it is used in practical situations.

Another interesting paper suggested in [7] shows a deep learning approach in which the authors relied on CNN based models to first remove the the background scene from hand gestures' images, extract the boundaries of the hands, and finally classify the gestures based on these boundaries. On the other hand, this approach has three main limitations:

- The impracticality of translating a hand's gestures to English letters rather than more useful words or phrases.
- The high latency due to the workflow of the suggested architecture where pictures are first captured, are stored, and then a CNN model is applied.
- The requirement of having a high-resolution image under specific lighting conditions for guaranteeing better results.

Compared to the previously mentioned papers, MAST has a higher accuracy compared to other Computer Vision-based proposed techniques as our approach uses Myo armband which accurately senses muscles' EMG signals regardless of the environment or lighting conditions. Moreover, the machine learning model utilized by MAST is an enhanced version of a dynamic random forest which does not need a large amount of data for training and getting accurate results.

On the other side of the spectrum, the usability of the previously proposed techniques is limited in the real-world and are only helpful for people who are familiar with a targeted sign language. In our approach, instead of focusing on a single limited language, we are trying to build a more comprehensive and high-accuracy general translation system which can be trained with any number and type of gestures of non-expert users. In our experiments, we focused on the medical field related gestures which can be used in smart hospitals equipped with Myo armbands, which can help any mute person to easily use the configured and trained gestures to converse with medical doctors.

III. MAST Implementation

A. Dataset Details

As mentioned previously, we focused on a medical field related hand gestures. Therefore, for this implementation, we chose 10 gestures from the American Sign Language (ASL) which has some medical meaning. Note that the Myo armband sensor does not have a gyroscope, thus we modified the original sign language for gestures which only consist of finger movements. Furthermore, to achieve a higher classification accuracy, we have stretched the fingers' motions compared to the original ASL gestures. As shown in Fig. 1. the gestures are comprehensible enough for the communication between patients and doctors (or nurses). Although the chosen gestures are translated into words rather than phrases, the patients can combine a subset of these gestures to converse with the doctors easily. For example, when a patient wants to express a "pain in the neck" to get a nurse's attention, (s)he can just use "Pain in neck" followed by "Nurse" gestures.

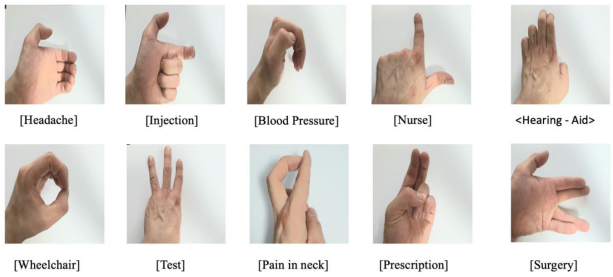

Fig. 1. 10 Gestures for Medical Sign Language

B. Random Forest Algorithm

The Random Forest (RF) algorithms form a family of classification methods that rely on the combination of several decision trees. The particularity of such Ensembles of Classifiers (EoC) is that their tree-based components are grown from a certain amount of randomness. Based on this idea, an RF is defined as a general principle of randomized ensembles of decision trees. Although an RF was developed in the 1990s, a formal definition of an RF was introduced in 2001 by Leo Breiman as follows [8]:

Theorem 1: A random forest is a classifier consisting of a collection of tree-structured classifiers $h(x, \Theta_k), k = 1, ..., L$ where x is an input vector and Θ_k's for $k = 1, ..., L$ are independent and identically distributed random vectors and where each k-th tree for $h(x, \Theta_k)$ casts a unit vote for the most popular class at input x.

In Breiman's RF definition, there are two randomization principles: Bagging and Random Feature Selection (RFS). Bagging is a training algorithm for an RF which applies the general technique of bootstrap aggregating to individual decision trees. Given a training set, bagging repeatedly selects a random sample of the training set and fits trees to these samples. RFS randomly selects for each of those trees a subset of features that will be withdrawn for the bagging operation.

An RF has several advantages for the classification of electromyogram data. First, an RF can train a model with a relatively small number of samples and get a higher accuracy compared to other algorithms. Due to the fact that building a large electromyogram dataset is hard, the classifier needs to perform classification with a small number of samples. Second, an RF has an effective method for estimating missing

data or features and maintains a high accuracy even if a large proportion of the data is missing. Third, an RF involves sampling of the input data with replacement, which is also known as bootstrap sampling, in the training phase. Thus, one-third of the original dataset is not used for training and can be used for testing. The testing set is called Out Of Bag (OOB). The whole algorithm's pseudocode to generate a classifier with *n* decision trees (estimators) is described in Algorithm 1.

Algorithm 1 Random Forest Algorithm

procedure CLASSICAL_RANDOM_FOREST(D)
  Generate a classifier with *n* estimators
  $i \leftarrow 1$
  while $i \leq n$ **do**
    Randomly sample the training data D with replacement to produce D_i
    Create a root node N_i containing D_i
    Call $Build\_Tree(N_i, D_i)$
    $i \leftarrow i + 1$
  end while
end procedure

procedure BUILD_TREE(N, D)
  if N contains instances of only one class **then**
    return
  else
    Randomly select x percent of the possible splitting features in N
    Select a feature vector F with the highest information gain by splitting
    Create f child nodes of $N_1, ..., N_f$ where F is a feature vector with f possible values $(F_1, ..., F_f)$
    $i \leftarrow 1$
    while $i \leq f$ **do**
      set the contents of N_i to D_i where D_i is all instances in N that match
      Call $Buil\_Tree(N_i, D_i)$
      $i \leftarrow i + 1$
    end while
  end if
end procedure

In Algorithm 1, $Build\_Tree(N, D)$ has the time complexity of $O(lg(n))$ and the while loop for the index i (from 1 up to n) in $Classical\_Random\_Forest(D)$ has the time complexity of $O(n \cdot v)$ without $Build\_Tree(N, D)$ since the loop is iterated n times and in each loop, v features are sampled from the dataset. Thus, the total complexity of Algorithm 1 is $O(v \cdot n \cdot lg(n))$.

C. MAST Architecture

The EMG data used in this paper is obtained by Myo armband. Myo armband is an electromyogram sensor with 8 sensing parts made by Thalmic Labs [9]. In our dataset, one EMG raw data is collected by Myo armband and hence fed to our algorithm which first finds distinct features of the raw data. These features are calculated through a linear discriminant analysis method with 6 mathematical formulas such as rms (root mean square), iav (sum of absolute value), ssi (square value), var (average of ssi), wl (sum of the distance between two adjacent EMG data), aac (average of wl) [3].

There are two parts of MAST such as Trainer and Translator Interfaces. We created a very simple and user-friendly trainer interface that can be used by non-expert users to train EMG data with any type and number of gestures. In the trainer section, EMG data is captured at 200Hz by $MyoDeviceListner$ which is offered by the official SDK. After capturing raw EMG data, a trainer calculates each gesture's EMG data features with formulas mentioned above and creates a csv file using Pandas, which is a Python data analysis library. In the translator section, a translator loads a csv file stored in the trainer section and fits the model with the data. Before fitting the data, However, EMG data is preprocessed for both boosting and normalization by an algorithm called $Adaboost$. After building the model, the translator captures the user's real-time raw EMG data and predicts the gesture. Fig. 2 shows the full architecture.

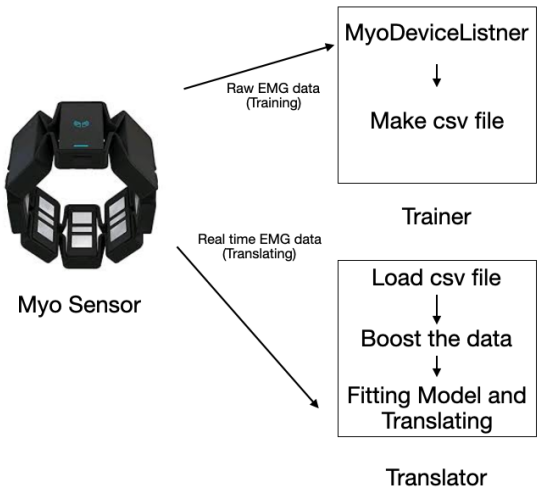

Fig. 2. MAST Architecture

D. MAST's Dynamic Random Forest

To improve performance, MAST uses a Dynamic Random Forest (DRF) method [10]. A classical Random Forest (RF) does not ensure that all trees contribute to the performance of a forest because it selects trees randomly. In contrast, a DRF guides the newly added tree to complement the existing trees as much as possible. That is, it is based on a sequential procedure that builds an ensemble of random trees by making each one of them dependent on the previous ones. In detail, a DRF chooses trees according to the predictions given by all the trees already added to the forest. The predictions are evaluated by the prediction ratio which is $W(C(pt, st))$ where $C(pt, st)$ is a contribution rate between the previous

trees (pt) and newly selected tree (st). This ratio is defined by:

$$C(pt, st) = \frac{1}{|D_{oob}|} \sum_{D_i \in D_{oob}} Count(st(D_i) == pt(D_i)). \quad (1)$$

In Equation (1), D indicates a sample of data and oob stands for Out-Of-Bag data. That is, we first create a set of called D_{oob} which consists of new data that have not been used in the generation of any decision tree existing in the random forest. These oob new data act as an indicator whether the newly selected tree st is useful or not. We can analyze the newly selected tree by counting how many times its prediction $st(D_i)$ for a sample D_i from D_{oob} matched the prediction of the previously selected trees in the forest $pt(D_i)$. The lower the value of $C(pt, st)$ is, the more the next tree will have to focus on the instance pt since it means that it was incorrectly classified by a large number of trees in the current forest. Consequently, the weight of st has to decrease with respect to $C(pt, st)$. In MAST's RF, we used the following weighting function: $W(C(pt, st)) = 1 - C(pt, st)$. With the DRF method, MAST's RF selects higher performance trees and attains a better accuracy compared to a classical RF which selects trees randomly without considering whether the added trees are useful or not. The whole algorithm of MAST's RF is described in Algorithm 2.

In Algorithm 2, $Build\_Tree(N, D)$ has the time complexity of $O(lg(n))$ if the time complexity of $C(pt, st)$ (i.e., $O(|D_{oob}|)$) in (1) is less than or equal to $O(lg(n))$. The while loop for the index i (from 1 up to n) in $MAST\_Random\_Forest(D)$ has the time complexity of $O(n \cdot v)$ without $Build\_Tree(N, D)$ since the loop is iterated n times and in each loop, v features are sampled from the dataset. Thus, the total complexity of Algorithm 2 is $O(v \cdot n \cdot lg(n))$.

Algorithm 2 MAST Random Forest Algorithm

procedure MAST_RANDOM_FOREST(D)
  Generate a classifier with n estimators
  $i \leftarrow 1$
  while $i \leq n$ **do**
    Randomly sample the training data D with replacement to produce D_i
    Create a root node N_i containing D_i
    Call $Build\_Tree(N_i, D_i)$
    $i \leftarrow i + 1$
  end while
end procedure

procedure BUILD_TREE(N, D)
  if N contains instances of only one class **then**
    return
  else
    Calculate $C(pt, st)$
    $W(C(pt, st)) = 1 - C(pt, st)$
    Select a feature vector F with the highest information gain by splitting
    Create f child nodes of $N_1, ..., N_f$ where F is a feature vector with f possible values $(F_1, ..., F_f)$
    $i \leftarrow 1$
    while $i \leq f$ **do**
      if $OOBTrees(pt) \neq \emptyset$ **then**
        $D_{i+1} = W(C(pt, st))$
      else
        $D_{i+1} = D_i$
      end if
      Call $Build\_Tree(N_i, D_i)$
      $i \leftarrow i + 1$
    end while
  end if
end procedure

IV. Performance Evaluation

A. MAST Random Forest vs. SCIKIT Random Forest

As mentioned previously in Section III-A, the gestures on which we evaluated our MAST are 10 medical gestures from the American Sign Language for which we compared the performance of our MAST's Random Forest against the state-of-the-art scheme such as SCIKIT's implementation. To carry out this comparison, we created a customized dataset for these gestures using the trainer interface mentioned in Section III-C, and to be more specific, we created a dataset by training each gesture for 30 iterations.

Fig. 3 shows the comparison between MAST and SCIKIT based on the number of the decision trees in the random forest, which is also known as the number of estimators. The reason we settled with a maximum of 25 estimators is due to the fact of no further accuracy improvement for both techniques. Noticeably, MAST achieves a higher accuracy than SCIKIT regardless of the number of estimators. That is, the low accuracy of SCIKIT can be explained by the way it works such that it selects all of its decision trees totally randomly. On the other hand, MAST achieves better results since it takes a preliminary decision before adding the tree to the random forest.

In addition, Fig. 4 shows another comparison based on the number of gestures to be classified. Although the accuracy between the two models is not differentiable in classifying a small number of gestures, MAST's accuracy is higher than SCIKIT in a greater number of gestures and has a negligible error. This result proves that MAST is a practical solution for practical scenarios where the number of classified gestures is big.

B. MAST Random Forest vs. CNN Model

To evaluate our MAST and benchmark it against another commonly used technique in sign language translation, an efficient deep learning model was designed to classify the same gestures. Since there are a lot of deep learning models for image classification, we chose one of the most popular models such as ResNet-18 along with applying SeNet to it [11], [12].

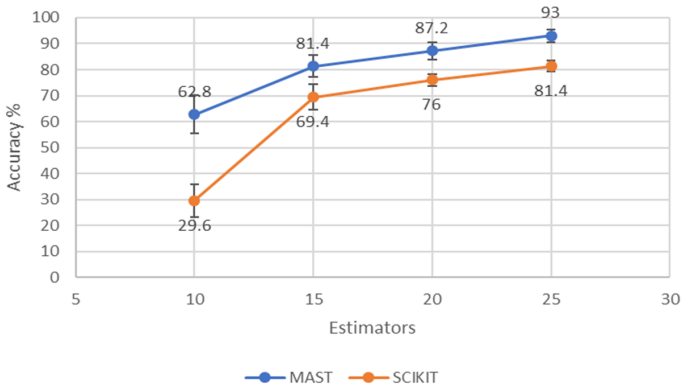

Fig. 3. Model Accuracy Based on the Number of Estimators

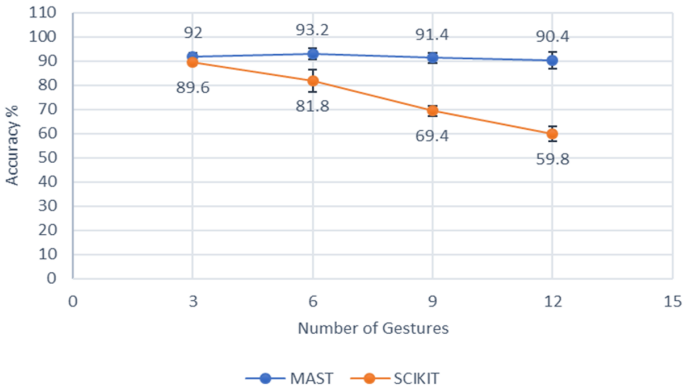

Fig. 4. Model Accuracy Based on the Number of Gestures

The data used for the experiment were the same gestures used to train MAST, but instead of EMG signals, images of the gestures were fed to the network. Each gesture class in the dataset had 300 images of size (112*112) and the whole dataset had a total of 3000 images. The model was trained using a single NVIDIA Tesla K80 and 24GB RAM. The validation results show that the model reached around 74.6% after 30 epochs. There are two distinct differences that make MAST clearly outperform the CNN implementation. First of all, MAST can achieve a top accuracy of 95% on the test data while the CNN implementation can only reach 74.6%. The second difference is training time, while MAST only needs 3∼6 seconds to train, the CNN implementation takes around 2 hours to train for 30 epochs on the machine specifications mentioned above. It might be argued that using a better GPU would decrease the training time, thus more training epochs can be performed. Although it may sound right, this is not the case. Our experimental results show that the CNN implementation cannot achieve higher than 74.6% accuracy regardless of the training time since it starts to overfit the training data, making the network perform worse on the validation data, as shown in Fig. 5. Note that the following figure ignores the error bars since the deviation in the results of all trials was less 0.9, therefore errors bars are negligible.

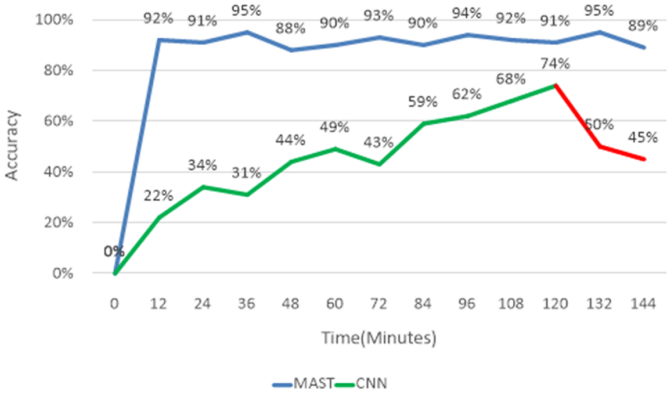

Fig. 5. MAST vs. CNN in Accuracy

C. MAST's Random Forest vs. SVM

To carry out this evaluation, we took the same approach that we used to compare MAST and SCIKIT. That is, we created a customized dataset for the gestures using the trainer interface, and hence we built a dataset by training each gesture for 30 iterations. Fig. 6 shows the gap in the performance between SVM which achieves a top accuracy of 58.8% compared to MAST's top 93% accuracy. Note that the state-of-the-art SVM implementation by SCIKIT was used and training for further time did not improve the accuracy for either of the techniques.

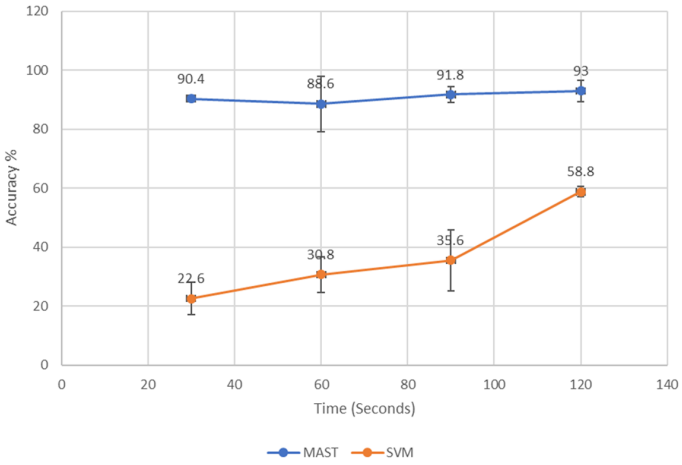

Fig. 6. MAST vs. SVM in Accuracy

V. Conclusion

This paper proposes a Myo Armband Sign-Language Translator (MAST), which is a novel algorithm for efficient sign language translation. MAST outperforms a popular deep learning technique such as CNN and a machine learning technique such as SVM. This is because MAST can accurately recognize different gestures with high generalization capacity when it is compared with CNN, SVM, and classical random forest techniques. However, since MAST can only recognize gestures

without taking into consideration the facial expressions which are important part of the sign language, the improvement of our work can be done by integrating face recognition models into MAST in order to provide a more accurate translation. Moreover, this opens a new door of research on not only using MAST in sign language translation but also utilizing the proposed dynamic random forest in any other machine learning regression or classification tasks. This is because MAST's random forest is marginally better than the popular SCIKIT's random forest. As future work, we will work on the fusion of MAST and a face recognition model for a better sign-language translation. Also, we will apply our dynamic random forest to the regression and classification in other domains.

ACKNOWLEDGMENTS

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2020-2017-0-01633) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation). This work was supported in part by IITP grant funded by the MSIT (No. 2019-0-01343, Regional strategic industry convergence security core talent training business). Note that Jaehoon (Paul) Jeong is the corresponding author.

REFERENCES

[1] M. Vernon, "Mental Health Services for People Who Are Deaf," *American Annals of the Deaf*, vol. 152, no. 4, pp. 374–381, Jul. 2007.

[2] C. Savur and F. Sahin, "American Sign Language Recognition system by using surface EMG signal," in *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, October 2016.

[3] S. Negi, Y. Kumar, and V. M. Mishra, "Feature extraction and classification for EMG signals using linear discriminant analysis," in *the 2nd International Conference on Advances in Computing, Communication, & Automation (ICACCA)*, October 2016.

[4] A. R. Babhulgaonkar and S. V. Bharad, "Statistical machine translation," in *the 1st International Conference on Intelligent Systems and Information Management (ICISIM)*, October 2017.

[5] Y. Madhuri, A. gOVINDHAN, and A. Mariamichael, "Vision-based sign language translation device," in *International Conference on Information Communication and Embedded Systems (ICICES)*, February 2013.

[6] M. Elmahgiubi, M. Ennajar, N. Drawil, and M. S. Elbuni, "Sign language translator and gesture recognition," in *Global Summit on Computer & Information Technology (GSCIT)*, June 2015.

[7] P. Mekala, Y. Gao, J. Fan, and A. Davari, "Real-time sign language recognition based on neural network architecture," in *the 43rd Southeastern Symposium on System Theory*, March 2011.

[8] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, October 2001.

[9] M. Sathiyanarayanan and S. Rajan, "Myo armband for physiotherapy healthcare: A case study using gesture recognition application," in *the 8th International Conference on Communication Systems and Networks (COMSNETS)*, January 2016.

[10] S. Bernard, S. Adam, and L. Heutte, "Dynamic Random Forests," *Pattern Recognition Letters*, vol. 33, no. 12, pp. 1580–1586, September 2012.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[12] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation Networks," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2018.