

A REVIEW OF CLASSIFICATION ALGORITHMS USED FOR GESTURE RECOGNITION USING ACCELEROMETER (2018)

¹Sudiksha Khanduja, ²J. Selvin Paul Peter, ³Sparsh Samir

^{1,3}Student, ²Assistant Professor,

^{1,2,3}Department of Computer Science, SRMIST, Chennai, India

¹sudiksha711@gmail.com, ²selvinpaulpeter.j@ktr.srmuniv.ac.in, ³sparshsamir1993@gmail.com

Abstract: Gesture recognition has the potent and capacity to be a natural means of interaction with PCs and can also prove to be a powerful tool in the domain of communication between the humans and the computers. A disparate range of algorithms are available in the areas of data mining, machine learning, gesture and pattern recognition for solving the same kind of problem. But there is a little guidance for suggesting which algorithm to use which gives best results for the problem at hand. Classification plays a vital role in many information administration and retrieval tasks. Document classification, also known as document categorisation, is the process of assigning a class to one or more predefined category labels. Classification is often posed as a supervised learning problem in which a set of labeled data is used to train a classifier which can be applied to label future examples [1]. Gesture classification includes different parts such as data processing, feature extraction, feature vector construction and final classification. Thus improvement in each part should lead to better results in document classification. In this paper, we apply machine learning methods for gesture recognition using classification techniques and algorithms available. The aim is to train our classifier by support vector machine (SVM) and K-nearest neighbour (KNN) algorithms. We aim to propose a framework for an accelerometer based gesture recognition of 26 letters of the english alphabet, all established and collected by means of a sensor placed on the palm of a person used as a device. Using machine learning the results achieved are recognitions made by 50 gestures, using the support vector machine classification methods with a performance of 0.94 accuracy. In Experiments, although both algorithms show acceptable results for gesture recognition, the performance of SVM is better in comparison to KNN.

Index Terms: Gesture Recognition, Accelerometer, Micro-controller, Bluetooth, Machine Learning, Classification Techniques, SVM, KNN.

1. Introduction

With the burgeoning diffusion of technology into the society, it is believed that the current modes of interactions with smart devices will become obsolete in

the coming years of information flow amongst the computers and the humans with rise in artificial intelligence and machine learning. PCs have turned into an indistinguishable part of our everyday life, affecting numerous parts of our day by day lives regarding correspondence and communication. Signals are a non-verbal type of correspondence which gives the HCI interface. Human Computer Interaction (HCI) amongst PCs and human comprehends human information sources and gives an easy to use interface to usability. As advances in innovation have been upgrading the personal satisfaction, more research has been coordinated towards characteristic HCI. The UI of a PC has advanced from a content based charge line to a graphical interface with console and mouse inputs. Be that as it may, they are badly designed and unnatural means of interacting with computers. The utilisation of hand motions gives an alluring other option to these unwieldy interface gadgets for human-PC cooperation (HCI). Client's by and large utilise hand signals for articulation of their emotions and notices of their contemplations. Specifically, visual translation of hand motions can help in accomplishing the straightforwardness and expectation wanted for HCI. The essential objective of signal acknowledgment explore is to make a framework which can distinguish particular English alphabets gestures and utilize them to pass on data or for gadget control.

This device will attain as a cross-platform and open-source python library which is exquisitely designed to make real-time gesture recognition more accessible using machine learning for the world. This library features a predictive support for building real-time gestures into transmissible, readable data. Users can hold the device to perform hand gestures with their preferred handheld styles and handwritings. Gestures in our system are divided into two types: the online gesture and the off-line gesture, which can be represented as a basic gesture sequence. The device is used to convert accelerometer data to a sequence of characters and sentences. The accelerations generated by hand motions are generated by the accelerometer embedded in the device, and are transmitted to a personal computer for further signal preprocess via the wireless module or a bluetooth module. Finally, the accelerations can be recognised via the alignment with the class templates in the testing stage.

The rest of the paper is organised as follows. First, in section 2, we give an introduction to classification method. In section 3, we discuss about feature selection and its related works. Subsequently, in section 4, we explain thoroughly our Persian web page classifier. In section 5, we describe our experiment and implied dataset and show the experimental results. Finally we conclude and outline future research direction in section 6.

2. Proposed Work

A. Accelerometer

An accelerometer is a micro electromechanical sensor (MEMS) that measures acceleration. It is also helpful in detecting tilt or orientation with respect to the earth of a device it is attached to [2]. This application uses its property to detect the tilt of the hand to move the mouse pointer. The continuous data streams are divided into individual gesture according to the A-button pressing label. An accelerometer can sense keenly the acceleration data of three spatially placed orthogonal axes in each gesture in a given sampling frequency. A gesture can be denoted as:

$$\mathbf{G} = (\mathbf{A}_x, \mathbf{A}_y, \mathbf{A}_z) \quad (1)$$

here $\mathbf{A}_x, \mathbf{A}_y, \mathbf{A}_z$ is the acceleration vector of an axis and L is the length of the temporal sequence. To describe the whole gesture but distinguish the periods from each other, we divide a gesture into $N+1$ segments identical in length, and then every two adjunct segments make up a **frame**. For the *feature extraction* of each gesture, firstly, it is divided into N frames. *Fourier transformation* is then employed on each frame, which uses an open source software package of FFTW. Features *mean*, *energy*, *entropy*, *correlation*, and *standard deviation* of individual axis in one frame are calculated using equation, combined as a feature vector. The feature vector is eventually put into a classifier in order to train a classification model or retrieve a recognised gesture type. More specifically, a gesture can be represented as: Intuitively, more frames a gesture is broken up into, more details we know about the gesture. However, it may lead to the over-fitting problem if the frame number N becomes large. It will also increase the dimension of the feature space, which increases computational complexity. We will conduct an experiment to determine the optimal frame number N later.

A. Micro-controller

Arduino is an open source platform that offers both hardware and software platform functionalities for

interactive projects that have the control and sensing capability providing an extension to connect to other such controls [3]. Arduino board uses a micro-controller board along with 14 digital input and output configuration pins and 6 analog inputs are used. The board is as shown in Figure.1. The output of the accelerometer sensor is given to three analog inputs of the Arduino board. The inbuilt ADC of the board converts the values to digital and sends the data to the Wireless or Bluetooth module through Serial Port. Arduino can be programmed through its IDE software that is built on Python.

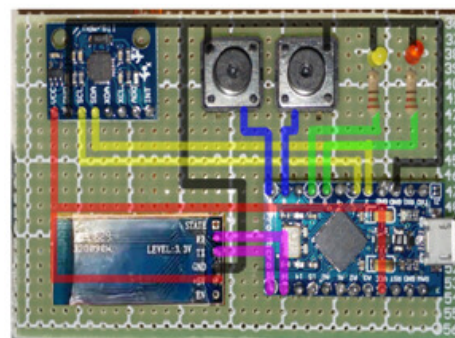


Figure 1. Circuit Connections

C. Bluetooth or Wireless Module

When someone wants to signal and record gestures we first press the button, and hence the module starts to send accelerometer data to the computer connected. At the moment when the button is released, the transmission stops immediately. The device can be powered by a power bank or by using a direct USB cable. Using a bluetooth HC-06 or a wireless module like SKETCH will help retrieve the data passed from the port of the micro controller.

D. Machine Learning

Gesture Recognition or any other machine learning tasks that uses supervised learning goes through a process of stages before the final classifier produces an efficient output. All the stages are indecent of each other and play a unique role in contributing to the accuracy of classification. Machine Learning techniques exploit organised data to create mathematical representations (models) of an observable phenomenon [3]. It is then possible to rely on such a model to provide predictions for unobserved data. In mathematical terms, let

$$\mathbf{s} = \{\chi_i \in \mathbf{R}^{1 \times p}, \mathbf{y}_i \in \mathbf{D}\}_{i=1}^n \quad (2)$$

be a training dataset [11]. In this formalism, n observations of a certain phenomenon are available: the i -th observation (or example) is characterised by p input features, constituting the vector χ_i , and a scalar target

value y_i that belongs to a certain domain D . Depending on the domain D , two types of ML problem are identified:

regression - if the output takes continuous values;

classification - if the output space is a set of classes or categories;

At a situation where we want to discriminate as to which class a gesture belongs to starting from the inertial data, the output belongs to the space of the known gestures. We are hence dealing with a classification problem. In the following we will indicate the number of different classes in the dataset with the notation N [4]. In earlier researches, known classifications algorithms have been adopted to gesture recognition problems, among which the most promising are those referring to Support Vector Machines (SVMs); in this work we have employed classification techniques (like SVM) that represent the state of the art in GR classification problems and compared these with KNN.

3. Classification Process In Machine Learning

Gesture Recognition or any other machine learning tasks that uses supervised learning goes through a process of stages before the final classifier produces an efficient output. All the stages are indecent of each other and play a unique role in contributing to the accuracy of classification.

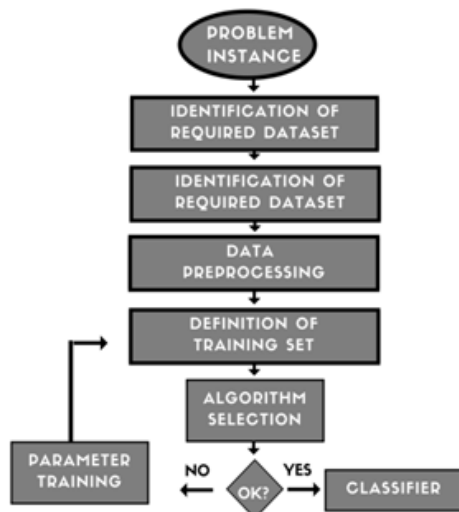


Figure 2. Machine Learning Architecture

A. Required Dataset:

The first step is to fabricate the data attributes that are crucial in determining the goal of the output. All values must have a constraint or an upper and lower bound limits to initiate remission of boundary limits. The data

set is then made available to be identified in terms of different parameters that can further extracted.

B. Data Pre-Processing:

Depending on the type of problem, the researchers have concluded that the pre-processing of data may vary on a disparate elements of attributes. For example, instance selection is not only used to handle noise but to cope with the infeasibility of learning efficiently from heavy datasets. Instance selection in such datasets is an optimization problem that aims to maintain the mining quality while attempts to degrade the sample size. It reduces data and enables a data classification algorithm to function and work effectively with very large or heavy datasets. There is a variety of procedures for sampling instances from a large dataset.

C. Training Set:

Once a clear input variable is made available on a global scope with logistic and constraint satisfying training inputs can be fetched at every input cycle, the similar categories of elements are classified together as a distinct entity. This is called as the training set and serves as the basis of classification space to make experiments and observe trends. The set consists of attributed key-value pairs of abundance different instances, that cumulatively make the set.

D. Algorithm Selection:

In this step, the documents are split into training and testing documents, the training documents are used to train the system (i.e. learn the system) to recognise different patterns of categories, the testing documents are used to evaluate the system, the process of categorisation depends on the algorithm used. The choice of which specific learning algorithm we should use is a critical step. Once we complete the preliminary testing and if achieved as to be satisfactory, the classifier that maps the unlabelled instances into classes is available for the routine to use. The classifier's evaluation is most often based on prediction accuracy (the percentage of correct prediction divided by the total number of predictions). There are three techniques used to calculate a classifier's accuracy which is essential One technique is to split all the training set by using two-thirds of it for training and the other third of it for estimating their performance. In cross-validation, the training set is divided into mutually exclusive, equal-sized subsets and for each subset the classifier is individually trained on the U of other subsets. The average of the error rate of each subset is roughly an estimate of the error rate of the classifier.

E. Evaluation:

The evaluation of the classifier or the categoriser is done by using many mathematic rules to test its effectiveness, such as recall, precision and F-measure. Recall (R) is defined as the ratio between the numbers of documents classified correct with respect to total number of document. Recall is defined as follows:

$$\text{Recall} = (\text{Number of positive predictions}) / (\text{Number of positive examples}) \quad (3)$$

Precision (P) is defined as the ratio between the number of documents classified correct with respect to the total number of documents classified. Precision is defined as follows:

$$\text{Precision} = (\text{Number of correct positive prediction}) / (\text{Number of positive prediction}) \quad (4)$$

Usually recall or precision alone are not used for the evaluation of the categorisation process because we can has height recall value but low precision value, so many researcher use measures such as F-measure which is a combination the standard recall and precision. F-measure combines recall and precision using the formula.

$$\text{F-Measure} = (2 * \text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision}) \quad (5)$$

4. Comparing Classification Algorithms For Gesture Recognition

A. Gesture Dataset :

In this section, we first elaborate on our data set and target values and then begin to evaluate the performances of two different algorithms for training of gestures. To improve precision we will create a new data set and evaluate their trends. Each time an accelerometer position is moved or a different accelerometer is used, the device becomes less accurate. So it is important to keep adding the training data to learning set in added time sequences. The algorithm can feature a maximum of 60 different gestures, such that each gesture is associated with a single letter of the English alphabet. The first step is to begin recording a new batch of data inputs for gestures by pressing the online button, the library records the data from movements of accelerometer using the following markup :

Target : Argument tells module it wants to record new gesture samples

A : It is the one that characterises the unique gesture of length 1

0 : Batch Number; It must be different each time we begin to

register a new batch in order to avoid overriding

Port : The port represents the serial port connection to the Micro-controller/Arduino

Ideally, each sample must be recorded and saved as a different file in the data folder and is later evaluated. Once the dataset is ready a suitable model is selected to train these inputs. The choice of a perfect algorithms can be a daunting and challenging task, hence we aim to compare 2 of the leading classification techniques on our dataset and evaluate the best model.

B. Classification Algorithms:

Classification algorithms use a set of variables such as “feature variable” or “independent variable” or “predictor variable” for making predictions and “class variable or dependent variable” [5][6] for holding outcome of prediction. These algorithms are driven by “class variable” for the given data [7] that can be either “categorical” or “continuous” [9]. “Classification” techniques are used in cases where the class label is categorical and “Regression” techniques are used in cases where the class label is continuous. A few examples of “categorical” variables are: Is person allergic or not, Will interest get defaulted or not, Is email spam or not, Does server fail or nor, Risk of getting cancer - High/Medium/Low. A few examples of “continuous” variables are prediction of change in temperature in next. The classifier selection is based on prediction accuracy of algorithm which is measured by a parameter called “Confusion matrix ” [8]. *Confusion matrix* provides information on false positives (fp), false negatives (fn), true positives (tp), and true negatives (tn) [9]. Confusion matrix is used to assess percentage of test data that has been correctly labeled by the classifier. [9]

$$\text{Accuracy: } (tp + tn) / (tp + tn + fp + fn); \quad (6)$$

$$\text{Precision: } (tp) / (tp + fp); \quad (7)$$

$$\text{Recall: } (tp) / (tp + fn); \quad (8)$$

1) Support Vector Machines (SVMs):

SVMs revolve around the notion of a “margin”—either side of a hyperplane that separates two data classes. Maximising the margin and thereby creating the largest possible distance between the separating hyperplane and the instances on either side of it has been proven to reduce an upper bound on the expected generalisation error. Support Vector Machine (SVM) is a popular classification technique discovered in 1995. A classification task usually involves separating data into training and testing sets. Each instance in the training set

contains one “target value” and several “attributes”. The final aim of the support vector machines is to make a model on the basis of training data which predicts the target values of test data by only giving out the attributes. Given a training set of instance-label pairs (x_i, y_i) , $i = 1, \dots, l$, this requires the solution of the optimization problem given as:

$$\min_{w,b,c} \quad \frac{1}{2}w^T w + c \sum_{i=1}^l \varepsilon_i \quad (9)$$

Here training vectors $X(i)$ are mapped into a higher (maybe infinite) dimensional space by the function ϕ . SVM finds a linear separating hyperplane with the maximal margin in this higher dimensional space. $C > 0$ is the penalty parameter of the error term. Furthermore,

$$K(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j) \quad (10)$$

is called the kernel function.

The following are four basic kernels:

Linear :

$$K(x_i, x_j) = x_i^T x_j \quad (11)$$

Polynomial :

$$K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0 \quad (12)$$

Radial Basis Function (RBF) :

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0 \quad (13)$$

Sigmoid :

$$K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r) \quad (14)$$

This algorithm needs that each data instance is represented as a vector of the real numbers. In case there are categorical attributes, it must be first converted into numeric data. Performing scaling before Support Vector Machine Algorithms is a very good practise. The very main advantage of scaling is to avoid all attributes in greater numeric ranges as to dominating those of them present in the smaller numeric ranges[11].

2) *k*-Nearest Neighbours (KNNs):

It is a simple algorithm that stores all available cases and classifies new classes based on a similarity measure (e.g. distance functions). From the early 1970's KNN has been used as a non parametric technique in the statistical estimation and pattern recognition.[10]. To work this classification algorithm out the steps are : By a majority vote of its neighbours a case is classified, most common amongst its K nearest neighbour measured by a distance function provided the case being assigned to the class. If $K = 1$, then the case is just assigned to the class of its immediate nearest neighbour.

There are three distance used for continuous variables: Euclidean (12), Manhattan (13), Minkowski (14).

$$\sqrt[k]{\sum_{i=1}^k (x_i - y_i)^2} \quad (15)$$

$$\sum_{i=1}^k |x_i - y_i| \quad (16)$$

$$(\sum_{i=1}^k (|x_i - y_i|^q))^{1/q} \quad (17)$$

$$D_H = \sum_{i=1}^k |x_i - y_i| \quad (18)$$

In instance of categorical variables, Hamming distance (15) must be used. It also brings up the issue of standardisation of the numerical variables between 0 and 1 when there is a mixture of numerical and categorical variables in the dataset. Choosing the best or the most optimal value for K is best achieved by prioritising to initially inspecting the data. A large K value is more precise as it reduces the overall noise but not necessarily error free. Cross-validation is another way to retrospectively determine a good K between 3-10.

5. Experiments And Results

The evaluation of the three algorithms can be also viewed from the False Rejection Rate (FRR) and False Acceptance Rate (FAR) aside from accuracy. FRR calculates the percentage of times the system produces a false reject, and FAR calculates the percentage of times the system produces a false acceptance. The graph showing the comparison of accuracy, FRR, and FAR for each algorithm on three different test cases are shown on Figure 3 – 4.

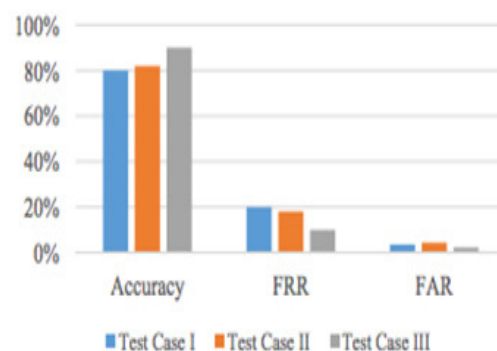


Figure 3 . Results for KNN

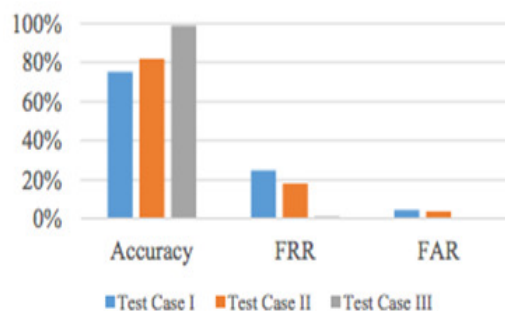


Figure 4 . Results for SVM

Both FRR and FAR showing a trend of decreasing when more data is given to the system. Low rate of FRR means that the percentage of the system make false rejection is low, which is good. Also, low rate of FAR means that the percentage of the system make false acceptance is low, which is also a good thing. The lowest FRR for each algorithm is 13.2% for KNN, 10% for SVM. The lowest FAR for each algorithm is 2.3% for KNN, 0.3% for SVM. Both the lowest FRR and FAR is the result of evaluation for Test Case III with the most data.

6. Conclusion

In this paper, we conduct a research comparing the performance of machine learning algorithm on classifying gestures. This study uses accelerometer based datasets as the source of signals and also sci-kit learn. The total of 26 variables from calculating feature distances we used as machine learning input for learning process. The result of discriminant analysis of the data shows 97.7% accuracy of classifying the data, while the machine learning algorithms perform well. The peak accuracy of SVM gets into 98.85%, for KNN it reaches 91.67%. The highest accuracy rate was reached when the largest amount of data was given, which is 60 instances of training data. The device is capable of successfully reading gestures in air to strings of computerise alphabet motions.

7. Application Domains

In this section, we would like to elucidate an overview on few of the many prominent areas of potential application for employ of a gesture recognition tool and framework for real-time applications.

A. Games:

If we look at the process of gesture conveyance in computer games: Konrad *et al.* [5] used varied gesture technology to control the movement of avatars in the virtual augmentation world, also Play Station 2 had come up with the Eye Toy, which is a camera that aims

to track movements for game player's hand or body [16]. Freeman *et al.* [6] tracked a player's hand or body position to control movement and orientation of interactive game objects such as cars. Games can have disparate applications for recognition of English alphabets in application and this framework will play its part being the unique available dataset for a well thrown and easy recognition of alphabets with just a swing of your hand in the air.

B. Virtual Reality:

With the increase in virtual reality and its booming gadgets, gestures have experienced an extreme level of uptake in computing applications. Virtual reality interactions use gestures to enable realistic manipulations of virtual objects using ones hands, for 3D display interactions [14] or 2D displays that simulate 3D interactions [15].

C. PC Applications:

In desktop computing applications, gestures can provide an alternative interaction to the mouse and keyboard [16]. Many gestures for desktop computing tasks involve manipulating graphics, or annotating and editing documents using pen-based gestures [7].

D. Home Control:

Use of smart character gestures can be used to control a smart environment or a smart space to ease the current methods of having to move to communicate and give inputs to electrical devices like buttons, switches, bells etc. In parallel to voice recognition, gesture recognition can also seek to become a potential means of communication for the advancements in technology.

References

- [1] X. Qi, B. D. Davison, Web Page Classification: Features and Algorithms, ACM Computing Surveys, 2007
- [2] Wen, T., Wang,L., Gu, J., Huang, B. "A 3-D Acceleration-based Control Algorithm for Interactive Gaming Using a Head-worn Wireless Device " 3rd International Conference on Bioinformatics and Biomedical Engineering , 2009.
- [3] T. Hastie, R. Tibshirani, and J. Friedman. The Elements of Statistical Learnin. New York: Springer, 2009
- [4] J. Wu, G. Pan, D. Zhang, G. Qi, and S. Li. Gesture recognition with a 3-d accelerometer. In

- Ubiquitous intelligence and computing, pages 25–38. Springer, 2009
- [5] David J. Pasta, “Learning When to Be Discrete: Continuous vs. Categorical Predictors,” *ICON Clinical Research*, San Francisco, CA
- [6] Isabelle Guyon, André Elisseeff, “An Introduction to Variable and Feature Selection,” *Journal of Machine Learning Research*
- [7] Yiming Yang and Jan O. Pedersen, “A Comparative Study on Feature Selection in Text Categorization,” *ICML ’97*
- [8] S. B. Kotsiantis, “Supervised Machine Learning: A Review of Classification Techniques,” *Informatica* vol 31, pp 249-268
- [9] Jiawei Han and Micheline Kamber, “Data Mining: Concepts and Techniques, 2nd ed.” The Morgan Kaufmann Series in Data Management Systems, Morgan Kaufmann Publishers
- [10] Duda, R. O., Hart, P. E., & Stork, D. G. (2012). *Pattern classification*. John Wiley & Sons.
- [11] Hsu, C. W., Chang, C. C., & Lin, C. J. (2003). A practical guide to support vector classification.
- [12] Whitehill, J., Serpell, Z., Lin, Y. C., Foster, A., & Movellan, J. R. (2014). The faces of engagement: Automatic recognition of student engagement from facial expressions. *Affective Computing, IEEE Transactions on*, 5(1), 86-98.
- [13] Teoh, T. T., & Cho, S. Y. (2011, July). Notice of Retraction Human emotional states modeling by Hidden Markov Model. In *Natural Computation (ICNC), 2011 Seventh International Conference on* (Vol. 2, pp. 908-912). IEEE.
- [14] Gandy, M., Starner, T., Auxier, J. & Ashbrook, D. “The Gesture Pendant: A Self Illuminating, Wearable, Infrared Computer Vision System for Home Automation Control and Medical Monitoring”. *Proc. of IEEE Int. Symposium on Wearable Computers*. (2000), 87-94
- [15] Konrad, T., Demirdjian, D. & Darrell, T. “Gesture + Play: Full-Body Interaction for Virtual Environments”. In: *CHI ’03 Extended Abstracts on Human Factors in Computing Systems*. ACM Press, (2003)
- [16] Website: University of Chicago: Mcneill Lab for Gesture and Speech Research. Electronic Resource, (2006)
- [17] Yiming Yang and Jan O. Pedersen, “A Comparative Study on Feature Selection in Text Categorization,” *ICML ’97*
- [18] M. Rajesh, Manikanthan, “GET-UP-AND-GO EFFICIENT MEMETIC ALGORITHM BASED AMALGAM ROUTING PROTOCOL”, *International Journal of Pure and Applied Mathematics*, ISSN NO:1314-3395, Vol-116, No. 21, Oct 2017.
- [19] SHUBHANSHU GUPTA, S. KOLANGIAMMAL, T. PADMAPRIYA, “Smart Curtain Using Internet Of Things” *International Innovative Research Journal of Engineering and Technology*, Vol. 2, Special Issue, pp. 4-8.

