An Approach for Recognizing Turkish Sign Language Characters with Gesture Control Device

[1]Zekeriya Katılmış, *[2]İlhami Muharrem Orak, [3]Alpaslan Duysak
[1] Tavşanlı Meslek Yüksek Okulu, Dumlupınar Universitty, Kütahya, Turkey
*[2]Faculty of Engineering, Department of Computer Engineering, Karabük University, Turkey
[3] İleri Teknolojiler Araştırma Merkezi, Dumlupınar Universitty, Kütahya, Turkey

Abstract

There has been several different approaches up to now to recognize sign languages. Usually camera images are used for this purpose. In this study we consider using a device which is developed mainly for virtual reality applications. Leap Motion is used for identifying finger gestures. The features of each sign character must be defined for recognition. In this study we introduced 30 different features based on hand, finger and joint points for identification of Turkish sign characters. K nearest neighbors and Naïve Bayes algorithms are used for classification. Sample signs are selected as “A” and “M” which are static posture symbols. High success rates are achieved in both methods.

Key words: Turkish Sign Language, character recognition, Leap motion, K nearest neighbor, Naive Bayes

1. Introduction

Sign language is an important tool for hearing impaired people to communicate with others. It is considered as native language of for those people. Hence, recognition of the sign language by means of machine is valuable for social respect as well as technological development.

All of the movements included in sign languages can be either posture or gesture type. Postures can be defined with single symbol whereas gestures are based on the change of the hand movements and shapes. Postures are used in finger spelling where each letter is shown one by one. On the other hand words or sentences are introduced by means of gestures.

In order to recognize the sign language with machine learning, the number of system is introduced. They can be mainly divided into two parts as glove based or image based according to the system used. In glove based system, some of the positions and angles of finger and hands are taken from the sensors located on the gloves and used for identification of the symbols. In a study, Artificial Neural Network (ANN) was used for classification of the data taken from a glove to recognition of Japanese sign language [1]. Similar study is conducted for Thailand language by taking data from 14 sensors and 4 movement tracking sensors and using ANN and segmentation techniques [2]. Although the quite high accuracy in glove based application, it is not so practical due to wearing need. For identification of sign language by using camera system, Starner et al. tried to recognize the words of ASL by applying Hidden Markov Model (HMM) to

*Corresponding author: Address: Faculty of Engineering, Department of Computer Engineering, Karabük University, 78050, Karabük, Turkey. E-mail address: imorak@karabuk.edu.tr, Phone: +903704332021 Fax: +903704333290

the segmented images for 40 different words. [3].

With the development of depth sensing with sensors, a new recognition system is emerged for hand movements. In this respect, Porfirio et al. used 3D and 2D representations of hand generated with Kinect sensors for Brasil Sign Language. Support Vector System (SVD) is used for the classification of characters [4]. In another work, an hybrid system including camera and Kinect sensors were used and the data is classified with SVM [5]

In the literature, there are also studies on the recognition of Turkish Sign Language (TSL). Some of these considered image processing similar to studies in other languages. Haberdar introduced real-time recognition system for TSL on video images using HMM and K nearest neighbor algorithm [6]. Memis used video images together with Kinect and classified the symbols using K nearest neighbour algorithm [7]. Multi Layer Prediction (MLP) and Radial Based Function are used by Göknar on 600 images taken from a camera [8]. For finger spelling in TSL, Altun et al. developed a system for static finger movement using Generalized Hough Transformation and scale-invariant feature transform (SIFT) [9].

In sign language recognition Leap-Motion has also been used in different applications. Nowicki et.al developed Leap movement library for static and dynamic finger recognition. They introduced a system for recognition using SVM, KNN and HMM [10]. Leap motion is also used for Arabic Sign Language (ArSL). [11]

In this study we introduced a system for recognition of Turkish Sign Language (TSL). A feature extraction is done for two posture characters. These characters are selected as “A” and “M”, which are performed with two hands together. Leap motion system which used in hand movement recognition is considered as the application system. Two different algorithms are considered for classification.

2. Classification Algorithms used In Application

For the recognition of character “A” and “M” , two different classification methods are used for. We considered using K nearest neighbor and Naïve Bayes algorithms for classification of TSL characters as they are used in most of the image classifications with high success rate [7, 11].

2.1. K Nearest Neighbor Algorithm

KNN is non parametric distance based algorithm. It is one of the most used algorithm in classification. Although it is simple, it produce very successful results. It consist of three stages as calculation of measurement function, selection of *k* value and classification of sample. Before classification, all data must be converted to numerical values. For a given test sample, distance to the training set are calculated and the nearest k number of samples are taken.

2.2. Naïve Bayes Algorithm

Naive Bayes algorithm uses some probability rules for classification. In this method, probabilities of each attribute being in each class are calculated. The probabilities of all attributes are multiplied and the most probable class is identified. Data are divided into two parts as training and test data. Assuming that all data we have are classified in one of the classes stated as $C_1, C_2, \ldots, C_n$. and the data consist of $X_1, X_2, \ldots, X_n$ different attributes. Any new value whose class is not known can be identified by Naive Bayes classifier. Among all calculated probabilities, the class having the highest value will be assigned as class for the new value.

3. Leap Motion Controller

Leap motion enable users to interact with software through hand movements. This equipment uses two cameras and infrared sensors to obtain the images of hands having depth information. It has 3D interaction environment in 8 ft^3 size with its large angle lenses. It tracks the finger, hands and similar objects. Information about hand and finger movements are gathered without delay for about 1/100 mm. Scanning 10 fingers with 290 fps can be achieved. It has the capability of 150° angle of field of view and Z axis depth. All the values are obtained in terms of mm value in real time.

Motion application programming interface (API) provides frames for tracking movement data. It generates continuously data and frames for the object tracked (Figure 1). If an object is recognized by the system, it is given a unique id and this id is used as long as the object is stayed in the view field.

Figure 1. Leap Motion tracking

4. Data Gathering and Feature Extraction

4.1. Features of Motion by Leap Motion

All the data produced by the Leap Motion is transferred to computer by means of USB port. In this study data transfer rate is defined as 10 fps. Data of each frame is not directly used instead mean of 10 frames are calculated. This improves the data accuracy of finger and hand movements obtained from the equipment.

Leap motion controller provides all movement data as an array. Data for hand movement consist of direction, position, speed, accuracy, capturing power, center of circumference and radius. Direction, position and speed features of palm show coordinate values in vector. Accuracy shows the accuracy of the data provided. Capture power defines whether the hand is closed or open. Accuracy, capture power and compression power value range between 0 and 1 depending on the status. Center of circumference and radius produce values for imaginary circle as if there is a ball inside the hand. Direction, position, length, distance, open-close status, finger order, fingertip speed, position, angle and distance of each finger bones and finger joints are considered as features of fingers. Direction and position of fingers are given in terms of coordinate data while distance and length are in millimeter. Coordinate values of distal, middle, proximal and metacarpal finger bones, distal interphalangeal joint (DIP), proximal interphalangeal joint (PIP), metacarpophalangeal joint (MCP) and fingertip (TIP) are given in vector form (Figure 2). Angle feature shows the angle between direction vector of fingers and x-z plane. Finger order defines the order of the finger relative to others in x-z plane. Finger speed produces millimeter value for finger tips in each second.

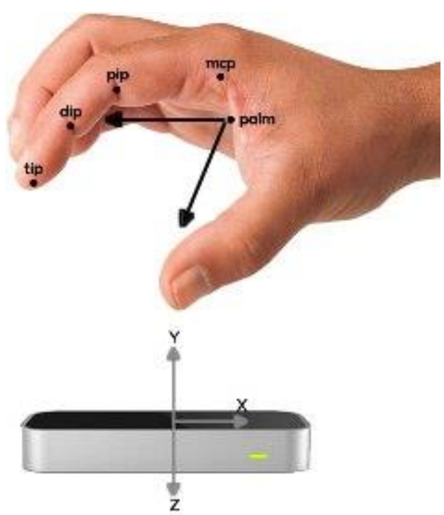

Figure 2. Leap Motion coordinate system and finger joints

4.2. Features Selected for Turkish Sign Language

Leap motion is designed for general purpose use. Considering the sign language recognition, not all of the features provided by leap motion may be needed. For example, features providing data on palm will not be used in this system. In this study we consider to select features more reliable and more meaningful for targeted symbols. Figure 2 a and b show “A” and “M” in TSL respectively. Following section describe features defined for these two characters. Although the features are selected for these two characters, feature extraction is considered for all TSL characters which will be covered in further study.

a) **b)**

Figure 2. a) Character A in TSL b) Character M in TSL

First of all, open and close feature of all fingers are extracted. If a finger is open it generate 1, otherwise 0. A sample code is given below.

frame.Hands.Leftmost.Fingers[0].IsExtended

Thumb, index finger and middle finger are mostly used in producing the characters with one or two hands. Hence the distance between their tip positions are used as identifying feature. These values are given in millimeters. 15 different features are specified based on distance between these three fingertip positions. The following code shows the obtaining feature data for distance between index finger and middle finger of left hand:

(frame.Hands.Leftmost.Fingers[1].TipPosition.DistanceTo(frame.Hands.Leftmost.Fingers[2].TipPosition))

Other than these features, character "A" has two distinct features. One of them is the distance between TIP position of the right thumb and PIP position of left middle finger. This can be obtain as;

(frame.Hands.Rightmost.Fingers[1].TipPosition.DistanceTo(frame.Hands.Leftmost.Fingers[2].JointPosition(Finger.FingerJoint.JOINT_PIP)))

The second feature is the distance between the position of proximal finger bone of right index finger and position of proximal finger bone of left index finger. A sample code for this one is given below.

(frame.Hands.Leftmost.Fingers[1].Bone(Bone.BoneType.TYPE_PROXIMAL).Center.DistanceTo(frame.Hands.Rightmost.Fingers[1].Bone(Bone.BoneType.TYPE_PROXIMAL).NextJoint))

For recognition of "M", additional three features are defined. One of them specifies the distance between right and left index finger. For that, PIP joints of the fingers are used. A sample code is given as *(frame.Hands.Leftmost.Fingers[1].JointPosition(Finger.FingerJoint.JOINT_PIP).DistanceTo(frame.Hands.Rightmost.Fingers[1].JointPosition(Finger.FingerJoint.JOINT_PIP)))*

For other two features, the distance between index finger and middle finger of right and left hands are considered. The distance for left hand can be obtained with

(frame.Hands.Leftmost.Fingers[1].Bone(Bone.BoneType.TYPE_INTERMEDIATE).NextJoint.DistanceTo(frame.Hands.Leftmost.Fingers[2].Bone(Bone.BoneType.TYPE_INTERMEDIATE).NextJoint))

Considering all above features which can be used for character "A" and M", all together 30 features are specified and used in classification algorithms.

5. Results and Discussion

In this study based on the defined features for character "A" and "M" two data sets are used. For training, total 100 data for each feature are produced as real time. For test purpose, another 100 data record are generated. These data are used to compare the performances of K nearest neighbour and Naive Bayes algorithms.

According to the tests, K nearest neighbour algorithm using Euclidean Distance correctly identify "A" character with 95% success rate. This ratio reaches to 100 % for "M". On the other hand Naïve Bayes algorithm produce 90% and 99% results for "A" and "M" respectively (Table 1).

Table 1. Classification Performance of the Algorithms

		K Nearest Neighbour			Naïve Bayes		
Character	Number of test	Correct Identification	False Identification	Success Rate	Correct Identification	False Identification	Success Rate
A	100	95	5	95 %	90	10	90%
M	100	100	0	100%	99	1	100%

Higher success rate for "M" is due to its more clear shape as well as its simpler nature. In implementation of "A", index finger of right hand cross over left hand fingers. This could be the reason for its less success rate. Crossing the fingers of two hands may result in data loss, data irregularities, or temporarily finger tracking stuck. This is also the case for "A". In K nearest neighbour algorithm, k is taken as 1. Since it produces high success rate no other values are tried.

Conclusions

In this study, two characters of TSL are considered for recognition by using leap motion system. Results show that both K nearest neighbour and Naïve Bayes algorithms produce very reliable identification. The selected characters are posture based characters. These obtained results are highly promising for future extension to gesture type characters. When the early results are compared with the results in the literature, better results are achieved in both methods. Although the latest version of Leap motion is used in development, there is still stability problem. In order to overcome that more features are needed to be used. Depending on the enhancement on the system, the number of features used in recognition system will be probably reduced in some extent in future applications. For this early study, K nearest neighbour and Naive Bayes algorithms produce effective results. However, for gesture type characters if same success rate cannot achieved, then some other algorithms will be consider as well.

References

[1] Murakami K, Taguchi H. Gesture recognition using recurrent neural networks. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: 27 April-2 May 1991; New Orleans. p. 237-242.

[2] Saengsri S, Niennattrakul V, Ratanamahatana CA.TFRS:Thai Finger-Spelling Sign Language recognition system. In Proceedings of Digital Information and Communication Technology and It's Applications (DICTAP): 16-18 May 2012; p. 457-462.

[3] Starner T, Pentland A. Real-time American Sign Language recognition from video using Hidden Markov Models. In Proceedings of International Symposium on Computer Vision, 21-23 November 1995; Coral Gabres. p 265-270.

[4] Porfirio AJ, Lais Wiggers K, Oliveira LES, Weingaertner D. LIBRAS Sign language hand configuration recognition based on 3D meshes. In Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (SMC), 13-16 Oct. 2013, p 1588-1593.

[5] Otiniano RK, Camara CG. Finger Spelling Recognition from RGB-D Information Using Kernel Descriptor. In Proceedings of Graphics, Patterns and Images (SIBGRAPI), 5-8 Aug. 2013; p 1-7.

[6] Haberdar, H. Saklı Markov Model kullanılarak görüntüden gerçek zamanlı işaret dili tanıma sistemi. MSc Thesis. Yıldız Technical University, Natural and Applied Science Institue; 2005.

[7] Memis, A, Albayrak S. Kinect RGB görüntülerde ve derinlik haritalarında uzam-zamansal özellikleri kullanarak Türk İşaret Dili tanıma. In Proceedings of Signal Processing and Communications Applications Conference (SIU), 24-26 April 2013; p 1-4.

[8] Goknar G, Yildirim T. Hand gesture recognition using artificial neural networks. In Proceedings on Signal Processing and Communications Applications Conference, 16-18 May 2005; p 210-213.

[9] Altun O, Albayrak S. Turkish fingerspelling recognition system using generalized hough transform, interest regions, and local descriptors. Pattern Recognition Letters 2011; 13:1626-1632.

[10] Nowicki M, Pilarczyk O, Wasikowski J, Zjawin,K. Gesture Recognition library for leap motion Controller, BSc Thesis, Poznan University Technology, Faculty of Computing, Institute of Computing Science; 2014

[11] Mohandes M, Aliyu S, Deriche M. Arabic sign language recognition using the leap motion controller, In Proceedings on IEEE 23rd International Symposium on Industrial Electronics (ISIE), 1-4 June 2014; p 960-965.