Real Time Gesture Detection and Recognition using Tensorflow.js

Anubhav Mishra[1] Shubham[2] Chirayu Aggarwal[3] Ranjeev Nayak[4]
[1,2,3,4]Department of Information Technology
[1,2,3,4]HMRITM, New Delhi, India

Abstract— Traditional methods of inputting data to systems or other digital gadgets to receive the desired outputs are no longer cutting-edge and progressive. Usable command set gets limited because of the limited usage of input devices such as a mouse. One of the solutions to these drawbacks is usage of gestures for inputting data and commands to the digital gadgets. In this paper we have presented an approach via which the devices become capable of detecting or recognizing the gestures and hence are able to perform the operations or tasks as desired by the users. Even in the surveillance systems, the gestures of humans are cogitated to be highly applicable information. In this paper we have discussed about gestures, it forms, its advantages and applications in today's world. We have made a model for the same using TensorFlow.JS Web library and have implemented Posenet algorithms. Anticipated method is implemented in Atom software.

Keywords: Gestures, Machine Learning, TensorFlow.js, Posenet, Gesture Sequence Normalization, Computational Complexity

I. Introduction

Gestures form a part of non-verbal communication. Gestures are very significant in expressing emotions of humans. In these the visible bodily activities perform communications and deliver the messages non-vocally. Movement of face, hands and other body parts form gestures. Expressive displays, proxemics etc. can effectively be communicated via gestures. A diversity of feelings and thoughts can be expressed with the help of these gestures. Gestures are generally a supplement with the speech. With the technological advancements growing progressively, novel techniques are being introduced every other day. One such novel technique is Gesture Detection or Gesture Recognition. It is a subject matter in computer science technology and aims at deducing the human gestures or postures using mathematical algorithms. Gestures commonly initiate from face or hands but they might even originate from any other bodily activities. Present emphases in this field comprise of emotion detection using hand or face gesture detection. Without making any physical touch to the device or digital gadgets, users can interact with these devices only by means of gestures. To deduce this sign language, many methodologies have been implemented using cameras and other algorithms of computer vision. However, it is important to note that detection and identification of these gestures are subject to the techniques we deploy for the same [1]. Detection of gestures in a way is seen as a means of computers and digital gadgets to be able to comprehend human body language. Thus, it is constructing a richer path between humans and machines. Input to the devices can now be made only with the help of a simple of a single action made by a finger or face. This action will make the point in the system to move accordingly. Figure 1 shows the basic working process of gesture detection technique.

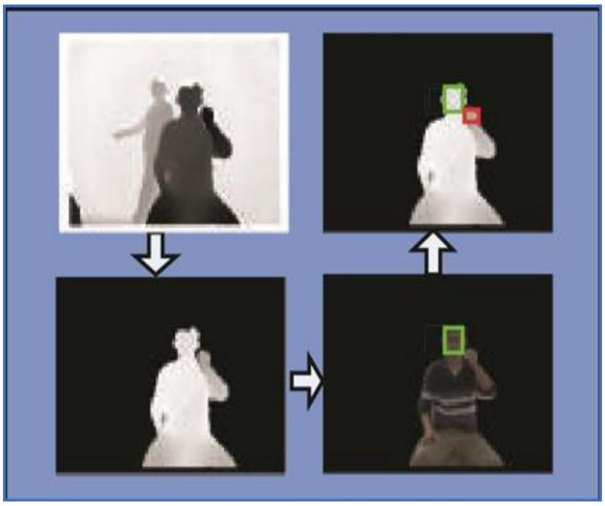

Fig. 1: Gesture Detection working process

This paper is organized as follows. Section II discusses about various advantages of using gesture detection and its applications. Section III showcases the various models employed in gesture detection. Section IV discusses particularly about hand recognition systems. The paper gets concluded in section V.

II. Advantages and Applications of Gesture Detection

Several features are provided by gesture detection technology. It is a highly accurate methodology and provides good stability. Some of the major benefits of gesture detection have been discussed here:

- Gesture detection technique has been said to be a hugely successful methodology because it helps in cutting down time in performing tasks. It saves time in device unlocking.
- Gesture recognition can be implemented with procedures from image processing and computer vision [2].
- Touchless Interface is an emerging trend seen in phones, laptops etc. Touchless user Interface (TUI) is the process of giving commands to the computer system with the help of gestures and bodily actions without using mouse, pen, keyboard etc.
- Paves way for an interaction which is powerfully spontaneous and also providing users with an enjoyable experience especially in systems such as gaming.

Gesture recognition finds its applications in diverse fields:

- Switching a channel without a remote:

Using hand gestures, we are able to switch channels, control audio, play games and also surf the internet.

- Automated Homes:

It enables us to control all the electronic devices of our homes via hand gestures. Gesture detection system can use the gestures like punching, kicking and waving our arms. It does not even a very massive set-up.

- Driving to safety:

Managing drive only using hand gestures. It finds its usage in blind-spot recognition and parking assistance. It is still not that widespread though.

– Automated sign language translator [3]:

It is now really easy to be able to convert our non- verbal signs into the textual format.

– Defence [4]:

At the battlefields, it is necessary to make complex tasks easier to perform. Gesture detection paves to be of utmost help in the remote areas to perform complex tasks effectively.

A. Chief Drawbacks of Gesture Detection Systems:

1) Telling Threat from Friend

The chief disadvantage of gesture detection systems is their incapability to tell friend from rival. Users within the system might have innocuous action flagged by the detection system, causing in a lock- down the net for an undecided time period till a technical expert can come on-site to recognise the issue and reorganize the recognition system. To a business reliant on fast activity for deadline orientated material, this can create a severe forfeiture of profit and client buoyancy, as associates might take business to a different place to an organization with a higher consistent network.

2) Discoverability

A setback with gestures, as previously found by Baudel and Beaudouin-Lafon in the year 1993, is that they are both not self-revealing and self- explanatory. A toolbar consisting of a named button has clear goal and easily detects gestures, nevertheless, might be random and are frequently more problematic to determine. To solve this issue, Bau and Mackay (2008) presented OctoPocus which is an active guide combining feedforward and feedback techniques. Following a press-and-wait gesture, a chart of all plausible gestures, envisaged via coloured templates, is showcased around the present position of the cursor. As the client initiates to track a path, the rest of the paths get gradually thinner, demonstrating that till the time they disappear, they are unlikely to be predicted.

3) Memorability

While traditional instructions have to only be diagnosed, gestures on the other hand require to be known and memorized before their execution as stated by Bau & Mackay, 2008. One option to generate memorable gestures is to build them as instinctive as possible, because they can easily be remembered in this manner (Wachs, Kölsch, Stern & Edan, 2011). Wobbrock et al. researched such natural gestures and observed that even though there are similar features used by almost every participant, gestures are no close to being palpable and that it is tough to devise a gesture set which is natural for each user. People frequently used gestures that are reversible to attain two conflicting effects and made use of more fingers to move larger objects, reflecting their knowhows in the real world. They also were powerfully prejudiced by their acquaintance of coventional computers, deploying gestures that could also be executed using a mouse and finding the gesture named as "Close" present at the objects' top-right corner just like they were making use of a Windows PC. (2009)

Table 1 represents the advantages and disadvantages of gesture detection system.

Advantages	Disadvantages
– Faster execution of tasks	– Telling threat from friend
– Touchless user Interface	– Not self-revealing and self-explanatory
– Can be implemented with procedures from image processing and computer vision	– Memorability is must
– Immediate and powerful interaction	
– Intuitiveness and enjoyability	

Table 1: Gesture detection advantages and disadvantages

III. MODELS FOR GESTURE DETECTION

There can be many different algorithms that can be implemented to model gesture detection. Liable to the kind of data input there can be many different algorithms used. Majority of techniques depend on 3D coordinate systems' key pointers. Gestures can be recognized with great accuracy contingent to quality of input and approach of algorithm. For interpreting actions and motions of body, they are classified in accordance with similar characteristics. In sign language, every gesture showcases a phrase or a word is an example for the same. [5] 'Toward a Vision-Based Hand Gesture Interface" discusses about the Human Computer Interaction. It describes many gesture systems that are interactive so as to capture entire gestures' space:

1) Manipulative
2) Semaphoric
3) Conversational

[6] An appearance-based and a 3D model-based approach are the two approaches being widely discussed in gesture detection arena. In appearance- based systems, videos and images are used for direct analysis. In 3D model-based systems, the 3D information about key elements of parts of body is used to get various significant parameters such as position of palms or joint angles.

In our research and project-making, we have used the appearance-based approach. We have used the Tensorflow.js library and have implemented Single and multiple posenet scenarios.

A. Tensorflow.js

It a library to perform mathematical computation.TensorFlow.js is a framework to define and run computations using tensors in JavaScript. A tensor is a generalization of vectors and matrices to higher dimensions. It develops ML with JavaScript. Use flexible and intuitive APIs to build and train models from scratch using the low-level JavaScript linear algebra library or the high-level layers API. Platform and environment: Tensorflow.js can run on browser and Node.js, and in both the platform there are many different available forms. Every platform has some unique factor that will affect the way it developed. In the browser, TensorFlow.js supports mobile devices as well as desktop devices. Each device has a specific set of constraints, like available WebGL APIs, which are automatically determined and configured for you. In Node.js, TensorFlow.js supports binding directly to the TensorFlow API or running with the slower vanilla CPU implementations. When a TensorFlow.js program is executed, the specific configuration is called the environment. The environment is comprised of a single global backend as well as a set of flags that control fine-grained features of TensorFlow.js. Here we have to use the Atom software for implementation of Gesture detection. Atom is a free and open-source text and source code editor for macOS, Linux, and Microsoft Windows with support for plug-ins written in Node.js, and embedded Git Control, developed by GitHub. Atom is a desktop application built using web technologies. Most of the extending packages have free software licenses and are community-built and maintained. Atom is based on Electron (formerly known as Atom Shell), a framework that enables cross-platform desktop applications using Chromium and Node.js. It is written in CoffeeScript and Less. It was able to be used as an integrated development environment (IDE), until that feature was 'retired' in December 2018. Atom was released from beta, as version 1.0, on 25 June 2015. Its developers call it a "hackable text editor for the 21st Century". And JavaScript has been used as backend of Atom. PoseNet is used for detection of posture detection. it is a pre-trained machine learning library that can estimate human poses. It was released by Google Creative Lab and built on Tensorflow.js. It's powerful and fast enough to estimate human poses in real time and works entirely in the browser. Even better, it has a relatively simple API.

B. *POSENET*

This package contains a standalone model called PoseNet, as well as some demos, for running real- time pose estimation in the browser using TensorFlow.js. PoseNet can be used to estimate either a single pose or multiple poses, meaning there is a version of the algorithm that can detect only one person in an image/video and one version that can detect multiple persons in an image/video. PoseNet is a machine learning model that allows for Real- time Human Pose Estimation. PoseNet is a machine learning model which allows for real-time human pose estimation in the browser, so what is pose estimation anyway? Pose estimation refers to computer vision techniques that detect human figures in images and video, so that one could determine, for example, where someone's elbow shows up in an image. To be clear, this technology is not recognizing who is in an image—there is no personal identifiable information associated to pose detection. The algorithm is simply estimating where key body joints are. PoseNet is a machine learning model that allows for Real-time Human Pose Estimation. PoseNet can be used to estimate either a single pose or multiple poses, meaning there is a version of the algorithm that can detect only one person in an image/video and one version that can detect multiple persons in an image/video. Single- person Pose Estimation: As stated before, the single- pose estimation algorithm is the simpler and faster of the two. Their ideal use case is for when there is only one person-centered in an input image or video. Multi-person Pose Estimation the multi-person pose estimation algorithm can estimate many poses/persons in an image. It is more complex and slightly slower than the single-pose algorithm but it has the advantage that if multiple people appear in a picture, their detected key points are less likely to be associated with the wrong pose. For that reason, even if the use case is to detect a single person's pose, this algorithm may be more desirable.

IV. HAND GESTURE RECOGNITION

Each gesture can be represented by a set of hand locations frame by frame, after hand tracking phase. The start and end frames of the gesture are manually annotated in this paper. In this section, we would describe how to pre-process the original gesture sequence and then a naïve/primitive shape descriptor, which is called shape order context, is introduced and we would also explain about how to combine two gestures using the proposed shape descriptor. At the end, we would compare the computational complexity between the proposed shape descriptor and the well-known shape descriptor: shape context.

A. *Gesture Sequence Normalization.*

Different gesture sequences have different lengths. However, for the same gesture sequences, the span of every two adjacent gesture points might be different due to the variation of the speed of sign. For the peer to peer points matching, we need to normalize each trajectory sequence so that they share the same point numbers. Specifically, gesture trajectories are linearly interpolated between every two adjacent original hand locations; then N locations are extracted equidistantly from all gesture points, so that each sequence has the same length. In our method, this sequences normalization step is significantly important because the shape-order context descriptor matching requires that each gesture sample has the same length and there is need to match the corresponding shape-order context descriptors of pair wise points in both training and testing data.

B. *Shape Context*

Shape context [7] is a descriptor which expresses the object shape by considering the relationship of the set of vectors originating from a point to all other sample points on a shape. Obviously, the full set of vectors as shape descriptors contains much too details since it configures the entire shape relative to the reference points. This set of vectors is identified as a highly discriminative descriptor which can represent the shape distribution over relative positions.

The shape context of pi is defined as a histogram hi of the relative coordinates of the remaining $n - 1$ point:

$$h(k) = \# \{q \neq pi : (q - pi) \in \text{bin } (k)\}$$

The bins are uniform in log-polar space, making the descriptor more sensitive to positions of nearby sample points than to those of points farther away.

C. *Our Approach: Shape-Order Context.*

A human hand dominated gesture trajectory can be simplified as a set of hand location features in spatial-temporal domain. For the traditional shape context method, the object is represented only in spatial domain. As a normalized gesture sequence with length L, we improve the shape context method by computing the relationship of a trajectory point to all other trajectory points in spatial-temporal domain.

Specifically, each k bin of the coarse histogram $h(\text{k})$ of the relative log polar coordinate is established by

accumulating sequence order difference between gesture points and remaining gesture points
Q= UjPj which is involved in the relative region.

In this way, the temporal information of gesture trajectory is embedded into log-polar space bins. Therefore, the value of each bin is dominated not only by the distance and the angle relation in gesture spatial domain, but also by impact by the sequence order relation in gesture temporal domain. Since the descriptors of shape order context contain rich spatial-temporal information for each point, they are inherently insensitive to small perturbations which are produced by different performers.

To build a robust gesture recognition system, translation and scale invariance are highly desirable. For our shape-order context approach, invariance under translation is internally existing since all measurements are taken with respect to points on the gesture trajectory.

To achieve scale invariance, we normalize all the gesture trajectory sets by means of calculating minimum enclosing circle (MEC) of those sets and then resize each set to the same circle's radius.

As for rotation invariance, since the shape order contexts are extremely rich descriptors and inherently insensitive to small rotation and perturbations of the shape, we use the absolute image coordinates to compute the shape order context descriptor for each point. Another reason for using the absolute image coordinates is that for two gesture trajectories even it achieves the same appearance after rotating; however, the complete rotation invariance impedes expressing the original meaning of the gesture. Hence, it should be emphasized that the completely rotation invariance of the gesture trajectory shape is not suitable for gesture recognition. Figure 2. Shows the l3og polar histogram bin semployed in shape- order contexts computation diagrammatical view.

D. Final Classification.

Due to the fact that each of the gesture sequence point is represented as histogram based on shape- order context, it is natural to use the x2 test statistic to compute the histogram similarity of pair wise (PiQi) in the two corresponding sequence positions. Thus, the cost Cij matching pair wise points not only includes the local appearance similarity but also contains local gesture order similarity, which is particularly useful when comparing the trajectory shapes. Finally, the similarity between two gestures is computed by accumulating each matching cost and using 1-NN nearest neighbour classification to determine which class the gesture belongs to. Datasets. Hand-signed digit datasets (HSD) [8] is another majorly deployed use hand gesture data set for gesture detection having 10 categories implemented through 12 distinguishable people

E. Computational Complexity.

The time complexity of the shape-order context matching algorithm can be measured as explained further. Let n be the number of gesture points after gesture normalization. Let r be the number of radial bins and let b be the number of angular bins. The time complexity of computing a gesture point histogram is n = r*a. Spruyt et al. [9] presents a whole tracking system providing long-term and real-time hand tracking along with unsupervised initialization.

The complexity of matching histogram in our shape-order context method is O(Pmn), where P is the number of gestures in the training dataset and mn is the feature vector size containing *m* points in each gesture trajectory.

Complete improved performance than one-dollar method [10] and Liu's method [11] that even device the sequence normalization along with changing sequence length is depicted.

Our proposed gesture recognition method has the same computational complexity as the original shape-order context algorithm. Table 2 depicts the comparison between each hand digital signature's false positive rate and the false negative rate.

Digital gesture	0	1	2	3	4	5
Proposed method						
FP (%)	0.052	0.074	0.052	0.0519	0.04	0.052
FN (%)	0	0	0	0	0.033	0
Liu's method						
FP (%)	0.51	0.51	0.51	0	0	0.25
FN (%)	2.27	9.09	0	0	6.82	0
One- dollar method						
FP (%)	0.048	0.170	0.048	0.033	0.070	0.056
FN (%)	0.033	0.033	0.067	0.467	0	0.033

Table 2: Difference between each hand digital signature's false positive and false negative rate depicting three measured methods.

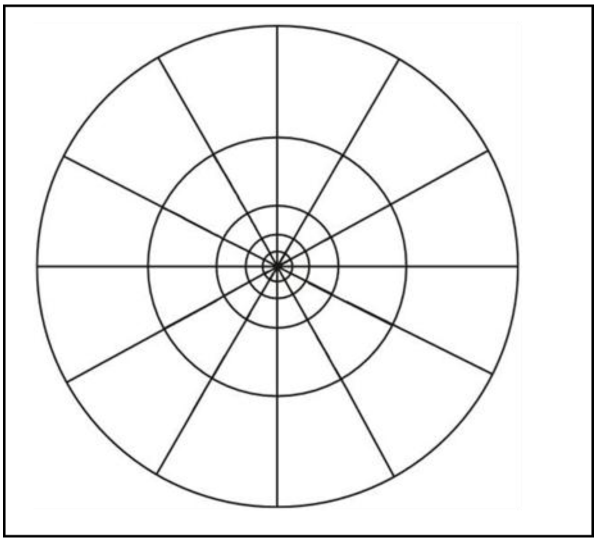

Fig. 2: Log polar histogram bin semployed in shape-order contexts computation diagrammatical view.

V. CONCLUSION AND FUTURE SCOPE

In this paper, we have showcased about the ongoing gesture recognition technology, its advantages and uses in the current industries. We have also talked about the particular hand gesture detection mechanism and its working. In the coming years, gesture detection technology's demand is going to rise to even greater extent. Our project has been implemented via tensorflow.js library and Posenet algorithm. We performed real-time human pose estimation with the help of the machine learning model i.e. Posenet. We implemented detections for both single and multiple poses using single posenet and multiple posenet algorithms respectively. Study relevant to manipulative, semaphoric and conversational gesture systems has been carried out.

Following which one category of gesture detection which is hand gesture detection has been presented. Based on the gesture sequence normalization, shape context i.e. consideration of the object's shape, we adopted shape-order context as the approach for estimating and detecting various hand gestures. The sequence point of gesture is presented as a histogram grounded on shape-order context. Lastly we showacsed the computational complexity associated with the shape-order context of the gesture under observation.

In the future, we tend to develop high stability working applications by deploying the same algorithms consuming machine learning as the technology. We aim at developing a gaming application that would recognize gestures. We intend to research about the gesture recognition for cultural-specific interactions adopting accelerometer-based gesture detection system in near future. It will deal with the user's conduct and one's elucidation of communications with others grounded on a person's cultural background.

References

[1] Matthias Rehm, Nikolaus Bee, Elisabeth André, Wave Like an Egyptian – Accelerometer Based Gesture Recognition for Culture Specific Interactions, British Computer Society, 2007

[2] Sultana A, Rajapuspha T (2012), "Vision Based Gesture Recognition for Alphabetical Hand Gestures Using the SVM Classifier", International Journal of Computer Science & Engineering Technology (IJCSET)., 2012

[3] Chai, Xiujuan, et al. "Sign language recognition and translation with kinect." IEEE Conf. on AFGR. Vol. 655. 2013.

[4] "Patent Landscape Report Hand Gesture Recognition PatSeer Pro". PatSeer. Retrieved 2017-11-02.

[5] Quek, F., "Toward a vision-based hand gesture interface" Proceedings of the Virtual Reality System Technology Conference, pp. 17-29, August 23–26, 1994,Singapore

[6] Vladimir I. Pavlovic, Rajeev Sharma, Thomas S. Huang, Visual Interpretation of Hand Gestures for Human-Computer Interaction; A Review, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997

[7] S. Belongi, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 4, pp. 509–522, 2002.

[8] http://vlm1.uta.edu/~athitsos/projects/digits/.

[9] V.Spruyt, A.Ledda, and W.Philips,"Real-time,long-term hand tracking with unsupervised initialization, "in Proceedings of the 20th IEEE International Conference on Image Processing (ICIP '13), pp. 3730–3734, Melbourne, Australia,September2013.

[10] J. O. Wobbrock, A. D. Wilson, and Y. Li, "Gestures without libraries, toolkits or training: a$1recognizer for user interface prototypes,"in Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology(UIST'07),pp.159– 168,2007.

[11] W.Liu, Y.Fan, T.Lei ,and Z.Zhang, "Human gesture recognition using orientation segmentation feature on random forest," in Proceedings of IEEE China Summit & International Conference on Signal and Information Processing(SIP'14),pp.480–484, Xi'an,China,July2014.