

---

# Trustworthy Artificial Intelligence and Automatic Morse Code Based Communication Recognition with Eye Tracking

---

Krishnakanth Medichalam<sup>1</sup>, V. Vijayarajan<sup>1</sup>, V. Vinoth Kumar<sup>2,\*</sup>,  
I. Manimozhi Iyer<sup>3</sup>, Yaswanth Kumar Vanukuri<sup>2</sup>,  
V. B. Surya Prasath<sup>4</sup> and B. Swapna<sup>5</sup>

<sup>1</sup>*School of Computer Science and Engineering, Vellore Institute of Technology,  
Vellore, India*

<sup>2</sup>*School of Computer Science Engineering and Information Systems, Vellore  
Institute of Technology, Vellore 632014, India*

<sup>3</sup>*Department of Computer Science and Engineering, East Point College of  
Engineering and Technology, India*

<sup>4</sup>*Department of Electrical Engineering and Computer Science, University of  
Cincinnati, OH 45221 USA*

<sup>5</sup>*Department of ECE, Dr MGR Educational and Research Institute, Chennai-95,  
India*

*E-mail: krishnakanthmedichalam@gmail.com; vijayarajan.v@vit.ac.in;*

*drvinothkumar03@gmail.com; srimanisen@gmail.com;*

*vyaswanth117@gmail.com; surya.iit@gmail.com; swapna.eee@drmgrdu.ac.in*

*\*Corresponding Author*

Received 06 April 2023; Accepted 04 August 2023;  
Publication 13 October 2023

## Abstract

Morse code is one of the oldest communication techniques and used in telecommunication systems. Morse code can be transmitted as a visual signal by using reflections or with the help of flashlights, but it can also be used as a

*Journal of Mobile Multimedia, Vol. 19\_6, 1439–1462.*

doi: 10.13052/jmm1550-4646.1964

© 2023 River Publishers

non-detectable form of communication by using the tapping of fingers or even blinking of eyes. In this paper, we develop a computer vision based approach that automatically characterizes the characters conveyed wherein a person can communicate to system or another person through Morse code with eye gestures. We can decode this visual eye tracking based language with the help of our automatic computer vision driven method. Our approach uses a normal webcam to detect the gestures made by the eyes and are interpreted as dots and dashes. These dots and dashes are used to represent the Morse code-based words. Image processing techniques-based blink and pupil detectors are employed. Blink detector helps us to detect a blink and the time that took for each blink. A blink that takes 2 to 4 seconds is acknowledged as a dot whereas a blink that takes more than 4 seconds is represented as a dash. The pupil detector helps us to detect the movement of the pupils, and if pupils move towards right with respect to a person then it is acknowledged as next letter and if the pupils are moved towards left with respect to a person then it is acknowledged as next word. In this way, we decode the Morse code which will be communicated using eyes and establish a non-detectable communication between a person and an automatic system. Our experimental results on an unconstrained visual scene with preliminary greeting words indicate the promise of an automatic eye tracking based system with success rate of 98.25% that can be of use in non-verbal communications.

**Keywords:** Morse code, communication, blinks, gestures, image processing, face detection, eye detection.

## 1 Introduction

Communication is a major problem for the people who are physically challenged. People who have vision, hearing impairments, people with speech disorders have difficulty in articulating sounds or signs to shape words. The paralyzed patients lose the ability to control their muscle movements and the only way to communicate is by using eyes. There exists nearly 15 million people in the world are suffering from speech disorders like brain damage, paralysis and many other diseases which makes the communication for those people a major issue. People with communication disabilities can sometimes use eyes and hand signs for primitive communications with others.

Eye blink detection and typing is an active area of research in the past few decades [4, 5, 8, 10]. Among a wide variety eye blink based research works we mention some of the relevant use cases. In [9] a system based on eye

blink detection with infrared sensor embedded in an eyeglass and conversion to Morse code was proposed. However, prolonged usage can be cumbersome. In [15] a device was developed which can act as a speech assistant where it can encode a Morse code to English text as a communication device and used by the people having disability. A blink detector and a microcontroller were used to develop the system. Blink detector consists of IR led and sensor and was used to detect the blinks of a person suffering from speech disorders by using the IR sensor values which differ when a person closes and opens eyes through which the author defined a short blink as a dot and a long blink as a dash. This electronic data is transferred to the microcontroller where the Morse code is converted to text and displayed on the LCD screen and also converted as voice output. In [6] Morse code based control method for a physically challenged person to control his/her wheelchair. It improved the driving efficiency for powered wheel chair communication called MCWN. The main challenging variables in the previous model are the continuous stops to counteract drift, exhaustion and irritation in challenging paths like slender hallways. A unique optimized code system was established that reduces the code length to bits with length two. The backward represents “.-“, the forward represents “..”, the left turn represents “-.-”, the right turn represents “-“. The dots and dashes are distinguished based on each switch press duration.

We know that when compared to the voice, the Morse code’s sensitivity is low for poor conditions of signaling, however humans can understand without any decoding devices, it is also used to construct speech when data was sent to listeners through voice channels. In [2] a system was proposed that decodes Morse code signals from a possibly noisy audio source, and display the decode text on LCD screen through a PIC microcontroller interface. The system consists of a handheld device that can receive Morse code through audio input or by a direct signal and translates it and displays on LCD screen. PIC microcontroller is used in this prototype to provide the interface for the input (Morse code) and output. Morse code as means of nondetectable communication was rewarding because the implementation of an event that has the controls of the communication is based on the time factor. In many of the disability issues at least a single gesture can be made. In terms of a single gesture, we can communicate through tongue gestures as studied in [13] and using image processing methods they decode the messages. In this work, some pre-defined gestures made by tongue and recorded through a webcam system were translated as dashes and dots of a double switched Morse code. A chart consisting of thirty characters has been created and

used as a search table. The position of tongue gestures is matched from the search table so the user need not to memorize the Morse code in advance. Among other possible techniques used for decoding Morse code to achieve the communication between physically challenged and the normal people are Morse code decoding using finger tapping [11], hand written code, etc.

In this work, we consider an implementation of a Morse code decoder using computer vision, in which Morse code is communicated using eye blinks alone. Physically challenged people can communicate through Morse code using eye blinks and the decoder will read the blinks and decode the message which will ease the communication for physically challenged people. Thus, this paper deals with similar type of issues and methods as in the aforementioned approaches from the literature, albeit with a different aim. We design and test an automatic computer vision based system that is developed for the physically challenged people who can use only eye blinks. We utilize image processing techniques such as the automatic face detection with active shape model which uses a template of the facial coordinates and matches the input data with the template data and produces results [3]. We then detect eye blinks and gaze direction through which the Morse code is received and decoded by the system. The main aim would be to increase the efficiency by increase the length of sentences which can be translated by the system. Our experimental results indicate the promise of approach in terms of smaller greetings words in Morse code implicating the usefulness of such a system in the information society context.

Regarding Automatic Morse code-based communication recognition with eye tracking, it is possible to develop systems that combine eye-tracking technology with AI algorithms to recognize and interpret Morse code input. Eye-tracking technology can track a person's eye movements and determine their focus of attention. By integrating this technology with AI algorithms, it becomes feasible to automatically detect and interpret Morse code signals based on a person's eye movements. The eye-tracking system can monitor the person's gaze and identify specific patterns or sequences associated with Morse code, while the AI algorithms can analyse and decode these patterns into meaningful messages. Such a system could be beneficial for individuals with motor disabilities or conditions that prevent them from using traditional communication methods. It would enable them to communicate effectively using Morse code by simply using their eye movements.

Rest of our work is organized as follows. Section 2 details our automatic eye tracking based Morse code based communication recognition. Section 3 provides our experimental results based on initial words and along with

accuracy of our system when compared with two different models. Finally, Section 4 concludes the paper.


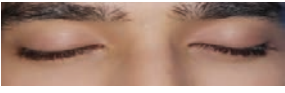
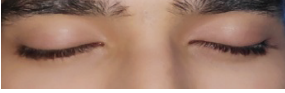


## **2 Methodology and Experimental Setup**

The Morse code decoder is implemented using python 3.7. We use Scipy, Imutils, Dlib, OpenCV and Numpy libraries for the decoder. First, we have used OpenCV to open the webcam for input. Then with the help of `dlib.get_frontal_face_detector()`. we detect the face. After the a library called `dlib.shape_predictor` comes into play for the detection of the eye area. Eye Aspect Ratio (EAR) is used to detect blinks and Horizontal Ratio (HR) is used to detect the pupil movement. Based on the threshold value of the EAR we detect dots and dahses. So, to get a dash as an output we have to blink our eyes once and close our eyes for 5 to 30 consecutive frames, and similarly, for a dot, we have to blink our eyes once and close our eyes for more than 30 consecutive frames. We interpret the right gazing as “#” which is acknowledged by the decoder that a letter is completed and next letter has begun and interpreted left gazing as “\$” which is acknowledged by the decoder that a word is completed. After giving input, we press the ‘q’ key to close the webcam. After clicking ‘q’ key the system exits and stores the dots and the dashes in the list which is the passed on to the machine learning models for the prediction of the characters. Once the prediction is done, we will then display the characters on the screen. This procedure is performed using a framework called flask which is used as an interface for the user. Table 1 shows the gestures and their respective representations.

### **2.1 Define Eyes Based Gestures**

Morse code is an international language in which each character is represented by using “dots” and “dashes”. We represent this code using eye gestures which are short blinks, long blinks, left gazing and right gazing. If a blink takes 2 to 4 seconds it is called a short blink and if a blink takes more than 4 seconds it is called a long blink. A short blink is acknowledged as “dot” and a long blink is acknowledged as “dash”. Right gazing is acknowledged as next character and left gazing is acknowledged as next word. We detect different gestures of the eyes which are converted into Morse code and then translated into text by an automated system. Thus, the four basic gestures utilized here are Short Blink, Long Blink, Right Gazing and Left Gazing.

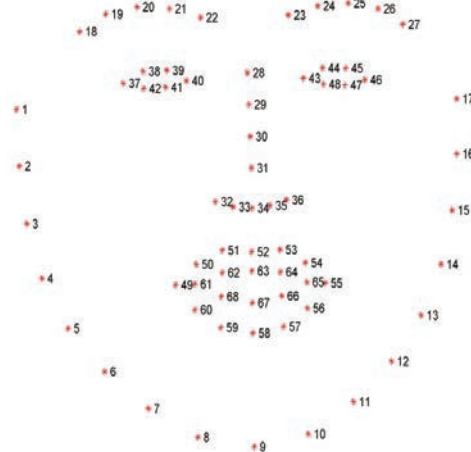
**Table 1** Gesture representations used in this work and their corresponding Morse codes and delimiters

Gesture	Representation
 Open Eyes	None
 Short Blink	Dot (“.”)
 Long Blink	Dash (“-”)
 Right Gazing	Next Letter (“#”)
 Left Gazing	Next Word (“\$”)

Our algorithm consists of two main steps:

1. Extracting Morse Code
2. Decoding Morse Code

For extracting the Morse code from eye blinking and gazing, we utilized computer vision techniques based on automatic input video capturing and processing from persons in a webcam setup. We used an efficient face detector that is based on the classical histogram of oriented gradients (HoG) feature combined with a linear classifier, image pyramid and sliding window detection approach [7]. The detection of eyes has been done using Dlib’s facial landmark detector.



**Figure 1** Representation of facial landmarks automatically computed from the video capturing of the face [3].

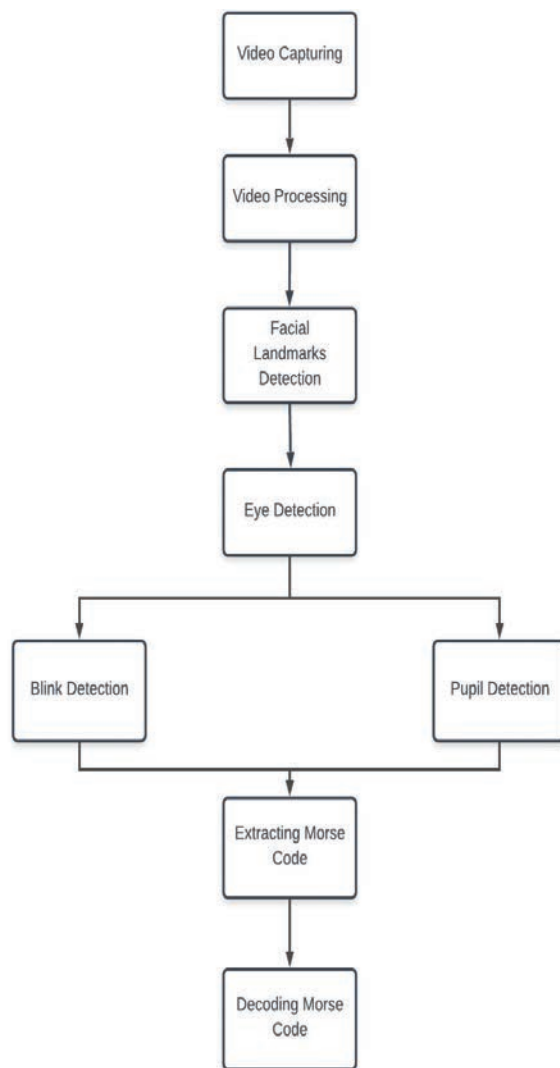
The facial coordinates can be used for the identification of the regions of face as shown in Figure 1. The pre-trained facial coordinate's detector which is present in the library named Dlib is used to estimate the location of all the coordinates of the face [12]. Using these landmarks, eyes are localized and represented for further processing. Further eye tracking approaches can also be utilized at this step [1]. After detecting the eyes we create a function which takes eye coordinates as inputs and returns us the coordinates of the pupil. This function first convert the eye frame into grey colored frame and removes the white part and leaves the black part which is generally called iris. The pupil is located on the centre of iris. Then, we calculate the centroid of the black part and return those coordinates called pupil coordinates. We take coordinates of both the eyes and use it in our algorithm for detecting blinks and pupil movements. Figure 2 shows the complete architecture of our algorithm.

## 2.2 Extracting Morse Code

This phase contains two main processes:

### 1. Blink Detection

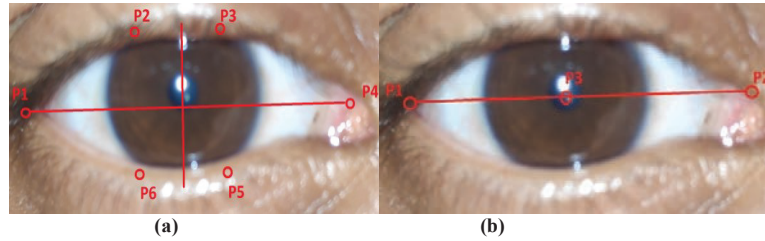
The traditional method for blink detection involves locating eye, finding white region of eye and when the white region disappear it recognize as blink. However, this method is less efficient and requires high computation. Hence,



**Figure 2** Overall flow of the Morse code based communication with eye tracking methodology considered here.

we utilize the eye aspect ratio (EAR), which gives us the ratio between facial coordinates of the eyes. Here, we represent each eye by 6 coordinates where 2 are horizontal and 4 are vertical as shown in Figure 3(a).

Whenever a person closes the eye this ratio becomes zero because the upper vertical coordinates join with lower vertical coordinates and distance



**Figure 3** Co-ordinates which are used in the (a) eye aspect ratio(EAR), and (b) horizontal ratio (HR).

become zero between those points, thus ratio also becomes zero. Let  $p_1$ ,  $p_4$  are horizontal coordinates and  $p_2$ ,  $p_3$  are upper vertical coordinates, and  $p_5$ ,  $p_6$  are lower vertical coordinates, then the EAR is calculated as,

$$EAR = \frac{|p_2 - p_6| + |p_3 - p_5|}{2p_1 - p_4} \quad (1)$$

Equation (1) is used in calculating a metric called the eye aspect ratio (EAR). It helps in calculating blinks in a much more refined way that involves a veritably simple computation rested on the ratio of distances between facial landmarks of the eyes. Here, the numerator represents sum of vertical distances and denominator represents horizontal distance. Since there are two set of vertical coordinates, and one set of horizontal coordinates we multiplied denominator with 2. This method based on EAR is more efficient, fast and easy to implement. After several trails we set the threshold value of this EAR as 0.3, whenever EAR is less than this threshold, for 5 to 30 consecutive frames it is acknowledged as dot, and for more than 30 consecutive frames it is acknowledged as dash.

## 2. Pupil Movement Detection

We have to show where the person is looking in realtime. So, we have to track their pupil movements. Our first step is to detect and mark the pupil correctly. Take the frame of both eyes separately and convert that frame into greyscale. Now the whole region is in black and white, the iris is black in color so remove white parts by binarizing the eye frame with suitable threshold. We find the optimized threshold by trial and error method. Now iris is located and we all know pupil is at center of iris. We pinpoint the pupil by finding the middle point of iris and draw contours. As pupil is located, now we have to find whether it is on left side or right side with respect to the person. For detecting which side the pupil is we use Equation (2).

It represents the horizontal ratio (HR) method. It provide a ratio which indicates the horizontal direction of the gaze, the extreme right is 0.0, the center is 0.5, and the extreme left is 1.0. We take two horizontal coordinates of eye  $p1(x1,y1)$ ,  $p2(x2,y2)$  and the pupil coordinates  $p3(x3,y3)$  as shown in Figure 3(b). The ratio of distance between one of the eye coordinates and the pupil coordinates to the distance between two coordinates of the eye gives HR, and is calculated as shown below.

$$HR = \frac{|p1 - p3|}{|p1 - p2|} \quad (2)$$

After extensive experiments, we set two thresholds for this HR that are 0.25 and 0.75. If HR is less than 0.25 then it is right gazing and if HR is greater than 0.75 then it is right gazing. Left gazing is acknowledged as “\$” and right gazing is acknowledged as “#”. These two (blink detection and pupil movement detection) models work simultaneously and store the outcomes into a string “S”. After this extracting process the output looks like  $S = “. \# . \$$ ”. Now we have to decode this string in to text format. Algorithm 1 explains the processes of this phase.

As we extracted the Morse code from the eye gestures, now we have to decode this Morse code into text. First, we create a Morse code dictionary where the keys of the dictionary are characters and their respective values are Morse code representation of that character. Initiate an empty string W. Traverse through the string S and add the characters to new string W until we encounter #. Now we have to check the dictionary for W and print its key. If we encounter \$ print space. The process has to be repeated until we complete traversing the string S till end. The output we get is the text representation of the string S. Algorithm 2 explains this phase.

### 3 Experimental Results

We evaluated the algorithm based on the accuracy and processing time by comparing our method with two other different methods that are: YCbCr and ViolaJones [16]. These methods are detect blinks. In YCbCr method the human face is detected first and then the eyes are extracted from the face using YCbCr color model, and in Viola-Jones method Haar like features are used for face localization and eye localization [14]. In both the methods the next step is template matching wherein the template of the face with eyes open is compared with the input template of the user’s face by which one can detect a blink.

---

**Algorithm 1** Extracting Morse code

---

**Result:** Extracted Morse code

Initialize Frame to 0;

Initialize an empty string S;

```

while eyes are detected do
  Get coordinates of eye frame;
  Calculate EAR;
  Detect pupil;
  Get coordinates of pupil;
  Calculate HR;
  if EAR is less than or equal to 0.3 then
    Frame =Frame + 1;
  else
    if Frame is greater than 5 and less than 30 then
      S=S+.
    else
      if Frame is greater than 30 then
        S=S+ ;
        Assign 0 to Frame;
      else
        end if
      end if
    end if
  end if
end while
if HR is less than or equal to 0.25 then
  S = S + #;
else
  if HR is greater than or equal to 0.75 then
    S = S + $;
  else
    end if
  end if
end if

```

---

**3.1 Comparisons**

$$ACCURACY = \frac{TP}{FN + TP} \quad (3)$$

$$FALSE\ ALARM\ RATE = \frac{FP}{FP + TP} \quad (4)$$

$$SUCCESS\ RATE = \frac{ACCURACY}{FALSE\ ALARM\ RATE + ACCURACY} \quad (5)$$

**Algorithm 2** Decoding Morse code**Result:** Decoded Morse code

Initialize dictionary D (Keys are Morse code and values are characters);

Initialize an empty string W;

```

for every character in S do
  if character is equal to # then
    Get the value of W from D;
    Print the value;
    Empty W;
  else
    if character is equal to $ then
      Print;
    else
      W = W + character;
    end if
  end if
end for

```

**Table 2** Performance of EAR method with frame rate less than actual

Trial	Blinks	TP	FP	FN	FAR (%)	SR (%)	Accuracy (%)	Time (sec)
1	5	5	0	0	0%	100%	100%	3.27
2	5	5	2	0	28%	78%	100%	3.13
3	5	5	1	0	16%	86%	100%	2.81
4	5	5	0	0	0%	100%	100%	3.34
5	5	4	0	1	0%	100%	80%	2.82
6	5	5	1	0	16%	86%	100%	3.15
7	5	5	0	0	0%	100%	100%	3.31
8	5	4	0	1	0%	100%	80%	2.94

Equation (3) represents the percentage of correct forecasts of the blinks, Equation (4) represents the percentage of incorrect deductions of the blinks and Equation (5) represents the percentage of correct deductions of the blinks. We calculated Accuracy, False Alarm rate, and Success rate(SR) for comparing the algorithms, where TP – true positive, FP – false positive, FN – false negative, TN – true negative. Further, that in this context, TP represents the blinks which are detected successfully while False positive represents the blinks which are wrongly detected and False Negative represents the blinks which are missed. Table 2 represents the results extracted using EAR method where we used this method only to detect blinks where the frame rate required for a blink to get acknowledged has been less than the actual frame rate that we used in our algorithm. Table 3 represents the results extracted

**Table 3** Performance of EAR method with actual frame rate

Trial	Blinks	TP	FP	FN	FAR (%)	SR (%)	Accuracy (%)	Time (sec)
1	5	5	0	0	0%	100%	100%	6.13
2	5	5	0	0	0%	100%	100%	7.26
3	5	5	0	0	0%	100%	100%	5.18
4	5	5	0	0	0%	100%	100%	4.91
5	5	5	0	0	0%	100%	100%	5.83
6	5	5	0	0	0%	100%	100%	4.87
7	5	5	1	0	16%	86%	100%	5.13
8	5	5	0	0	0%	100%	100%	5.78

**Table 4** Performance of YCbCr method

Trial	Blinks	TP	FP	FN	FAR (%)	SR (%)	Accuracy (%)	Time (sec)
1	5	5	2	0	28%	78%	100%	2.8
2	5	5	2	0	28%	78%	100%	3.4
3	5	5	2	0	28%	78%	100%	2.9
4	5	2	2	3	50%	44%	40%	2.3
5	5	5	2	0	28%	78%	100%	2.0
6	5	5	2	0	16%	83%	83%	3.1
7	5	5	1	0	16%	83%	83%	3.2
8	5	2	2	3	50%	44%	40%	2.0

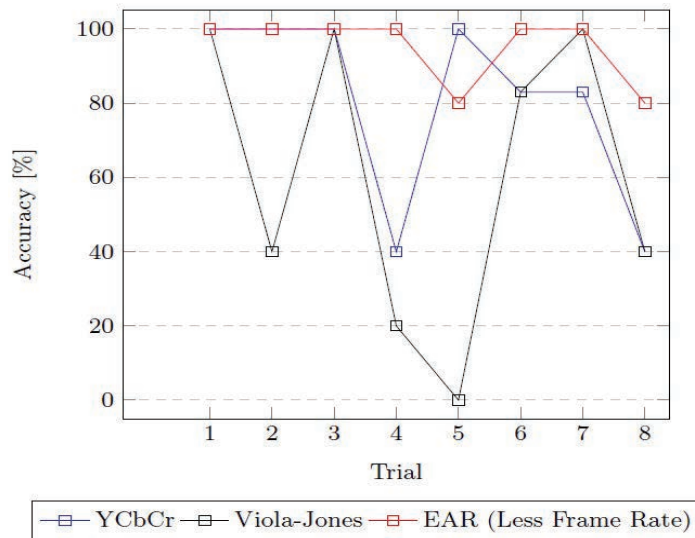
**Table 5** Performance of Viola-Jones method

Trial	Blinks	TP	FP	FN	FAR (%)	SR (%)	Accuracy (%)	Time (sec)
1	5	5	2	0	28%	78%	100%	3.9
2	5	2	2	3	50%	44%	40%	4.0
3	5	5	2	0	28%	78%	100%	6.1
4	5	1	2	4	66%	16%	20%	3.7
5	5	0	0	5	0%	0%	0%	4.2
6	5	5	1	0	16%	83%	83%	5.1
7	5	5	0	0	0%	100%	100%	5.3
8	5	2	2	3	50%	44%	40%	4.8

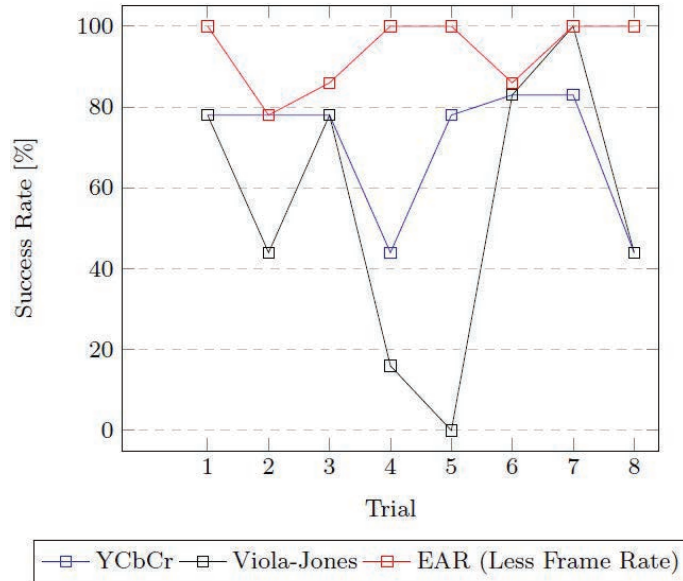
using EAR method where we used this method to detect blinks where the frame rate required for a blink to get acknowledged is same as the frame rate that has been used in our algorithm. Our EAR method obtained better performance overall across eight subjects when compared with the YCbCr method in Table 4, and Viola-Jones method in Table 5. Moreover, the EAR method gives us higher success rate and accuracy as shown in Table 6. However, the processing time for each blink is slightly higher compared to

**Table 6** Comparison of different methods in terms of accuracy, processing time and success rate. In our model, < EAR (frame rate less than actual rate) , = EAR (actual frame rate)

S.NO	Average Values	YCbCr	Viola Jones	<	=
1	Accuracy	80.75%	60.37%	95%	100%
2	Processing time (per blink)	0.38	0.65	0.61	1.12
3	Success rate	71%	55%	93.75%	98.25%

**Figure 4** Comparison of accuracy between YCbCr, Viola-Jones and EAR (less frame rate).

the YCbCr method that takes 0.38 secs, and the Viola-Jones method takes 0.65 secs while our method takes 0.61 secs which is better when compared to ViolaJones method. In terms of accuracy the YCbCr method gives 80.75%, Viola-Jones method gives 60.37% while our method with frame rate less than the actual frame rate gives 95%. In terms of success rate the YCbCr method gives 71%, Viola-Jones method gives 55% while our method with frame rate less than the actual frame rate gives 93.75%. Our method has proven to be the better one when compared with the two methods in terms of accuracy and success rate. The EAR method with actual frame rate that is used in our algorithm has processing time of 1.12 secs, Accuracy of 100%, and Success rate of 98.25%. Here we have increased the frame rate to achieve 100% accuracy so that the algorithm could not detect the normal blinks and only detect the blinks that are intentional for representing a character. So, the processing time has been increased. Figures 4 and 5 shows the Accuracy and



**Figure 5** Comparison of success rate between YCbCr, Viola-Jones and EAR (less frame rate).

Success Rate trends by our EAR based method along with the two methods in different trials indicating the superiority of our app.

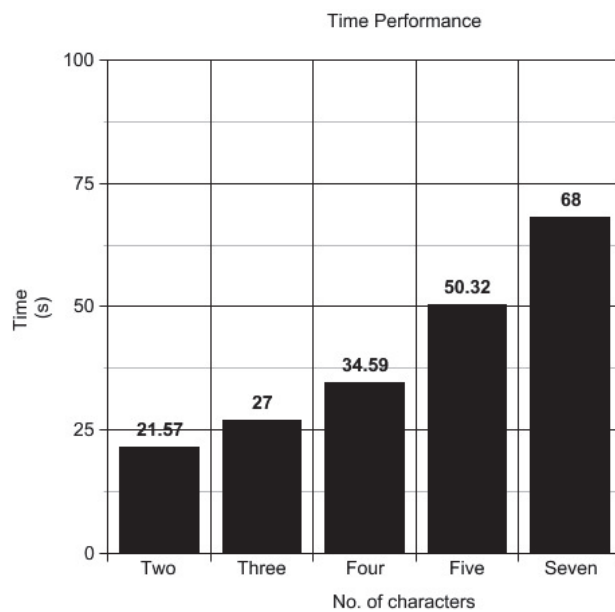
The developed algorithm was tested by eight people where each person tested the algorithm with seven different texts having different length. The texts that were tested are: HI, YO, HEY, HER, WHAT, HELLO, HEY YOU. The subjects were given the Morse code of abovementioned texts and asked to represent them using eye gestures. The algorithm captured the gestures and converted back into text format. Table 7 represents the time taken by one of the subjects to perform the gestures required for the given texts. To analyze the time performance by the algorithm we calculated the average time for a text to perform by a subject as shown in Table 8. We also calculated the standard deviation for each text. It has been observed that there has been a very less deviation between each subject. Figure 6 represents the performance of our algorithm with different number of characters. Figures 7 shows the working of our eye-tracking based application interface with typical examples corresponding to the Morse Code. Our results indicate that the eye blinking and gazing based automatic Morse code based communication recognition is possible, however, deeper analysis with implementing on a multi-person detector is needed for further validation.

**Table 7** Time taken by one of the subjects across three trials

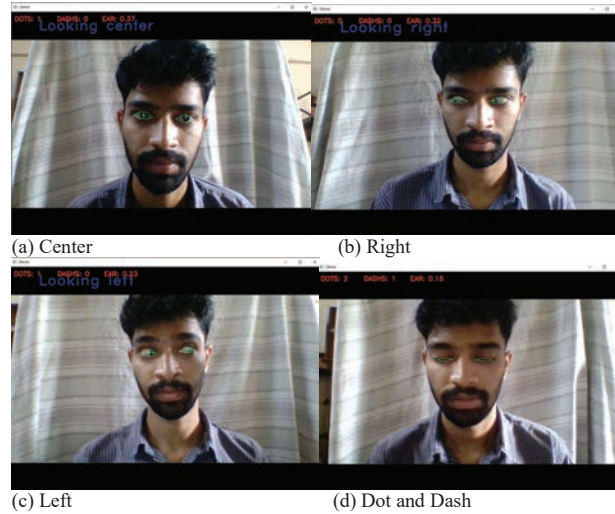
Text	HI (Secs)	YO (Secs)	HEY (Secs)	HER (Secs)	WHAT (Secs)	HELLO (Secs)	HEY (Secs)	YOU (Mins:Secs)
Trail-1	16.40776	28.35784	31.12828	27.38033	36.32479	51.14716	51.14716	01:09.51930
Trail-2	15.94406	24.53599	28.64970	21.83890	34.15836	49.69053	49.69053	01:07.67951
Trail-3	18.78828	25.40792	30.34583	22.69795	33.29166	50.13936	50.13936	01:06.82710

**Table 8** Overall average time (s) taken for different words. SD – standard deviation

Text	Range (s)	Average(s)	SD
HI	15.94 – 18.78	17.0467	±1.245
YO	24.53 – 29.35	26.1005	±1.633
HEY	28.64 – 31.12	30.0412	±1.039
HER	21.83 – 27.38	23.9723	±2.436
WHAT	33.29 – 36.32	34.5916	±1.247
HELLO	49.69 – 51.14	50.3256	±0.607
HEY YOU	66.82 – 69.51	68.0086	±1.123



**Figure 6** Representing time performance in terms of the number of characters.



**Figure 7** Example eye tracking with gazing (a) center, (b) left, (c) right, and (d) dots and dashes

## 4 Conclusion

A computer vision based character recognition algorithm using eye blinks has been proposed in this paper. We have effectively demonstrated our algorithm only by using image processing techniques which are robust enough to detect blinks and pupil's movement. Instead of using complex methods that can be computationally heavy to detect eyes, here we used a fast facial landmark detector based approach which made use of less computational power with efficient results. Unlike traditional methods for blink detection that involve locating eye, finding white region of eye and blink recognition based on disappearing white regions of the eye, our model uses eye aspect ratio (EAR) concept directly from the face landmark extraction which yielded desired results with low computation power requirements. For pupil movement detection we used horizontal ratio (HR) method which has been proven sufficient enough to get accurate results. The method considered here need to be tested on a larger group of people communicating with each other instead of individual persons tested here. Further, utilizing depth cameras with 3D information along with our 2D EAR and HR can obtain more accurate results. The only drawback in this system is that it takes a good amount of time if a person wants to represent a long sentences using dots and dashes. So, in the future, we can enhance the model by getting rid of the time constraints

and applying cutting-edge machine learning techniques like neural networks. This model could be expanded to encompass paragraphs, words, sentences, and even complete sentences.

## References

- [1] Al-Btoush, A.I., Abbadi, M.A., Hassanat, A.B., Tarawneh, A.S., Hasanat, A., Prasath, V.S.: New features for eye-tracking systems: Preliminary results. In: 2019 10th International Conference on Information and Communication Systems (ICICS), pp. 179–184. IEEE (2019). DOI: 10.1109/IACS.2019.8809129.
- [2] Bakde, N., Thakare, A.: Morse code decoder using a pic microcontroller. *International Journal of Science, Engineering and Technology Research (IJSETR)* 1(5), 200–205 (2012).
- [3] Bidani, S., Priya, R.P., Vijayarajan, V., Prasath, V.: Automatic body mass index detection using correlation of face visual cues. *Technology and Health Care* 28(1), 107–112 (2020). DOI: 10.3233/THC-191850.
- [4] Cheng, X., Zhao, X., Zhou, J., Zhang, H.: Eye localization method based on contour detection and d-s evidence theory. *Signal, Image and Video Processing* 12(4), 599–606 (2018). DOI: 10.1007/s11760-017-1165-9.
- [5] Fathi, A., Abdali-Mohammadi, F.: Camera-based eye blinks pattern detection for intelligent mouse. *Signal, Image And Video Processing* 9(8), 1907–1916 (2015). DOI: 10.1007/s11760-014-0680-1.
- [6] Ka, H., Simpson, R.: Effectiveness of morse code as an alternative control method for powered wheelchair navigation. In: *Proceedings of the Annual Conference on Rehabilitation Engineering*. RESNA (2012).
- [7] Venkatesan, V.K., Ramakrishna, M.T., Izonin, I., Tkachenko, R., Havryliuk, M. Efficient Data Preprocessing with Ensemble Machine Learning Technique for the Early Detection of Chronic Kidney Disease. *Appl. Sci.* 2023, 13, 2885. <https://doi.org/10.3390/app13052885>.
- [8] Devarajan, D., Alex, D. S., Mahesh, T. R., Kumar, V. V., Aluvalu, R., Maheswari, V. U., and Shitharth, S. (2022). Cervical Cancer Diagnosis Using Intelligent Living Behavior of Artificial Jellyfish Optimized With Artificial Neural Network. *IEEE Access*, 10, 126957–126968. <https://doi.org/10.1109/access.2022.3221451>.
- [9] Mukherjee, K., Chatterjee, D.: Augmentative and alternative communication device based on eye-blink detection and conversion to morse code

- to aid paralyzed individuals. In: International Conference on Communication, Information & Computing Technology, pp. 1–5. IEEE (2015). DOI: 10.1109/ICCICT.2015.7045754.
- [10] S. T. Ahmed, V. Kumar and J. Kim, “AITel: eHealth Augmented Intelligence based Telemedicine Resource Recommendation Framework for IoT devices in Smart cities,” in *IEEE Internet of Things Journal*, DOI: 10.1109/JIOT.2023.3243784.
- [11] Ravikumar, C., Dathi, M.: A fuzzy-logic based morse code entry system with a touch-pad interface for physically disabled persons. In: 2016 IEEE Annual India Conference (INDICON), pp. 1–5. IEEE (2016).
- [12] Sagonas, C., Antonakos, E., Tzimiropoulos, G., Zafeiriou, S., Pantic, M.: 300 faces in-the-wild challenge: Database and results. *Image and Vision Computing* 47, 3–18 (2016). DOI: 10.1016/j.imavis.2016.01.002.
- [13] Sapaico, L.R., Sato, M.: Analysis of vision-based text entry using morse code generated by tongue gestures. In: 2011 4th International Conference on Human System Interactions, HSI 2011, pp. 158–164. IEEE (2011). DOI: 10.1109/HSI.2011.5937359.
- [14] Singh, H., Singh, J.: Real-time eye blink and wink detection for object selection in HCI systems. *Journal on Multimodal User Interfaces* 12(1), 55–65 (2018). DOI: 10.1007/s12193-018-0261-7.
- [15] Srividhya, G., Murali, S., Keerthana, A., Rubi, J.: Alternative voice communication device using eye blink detection for people with speech disorders. *International Journal of Recent Technology and Engineering* 8(4), 12,541–12,543 (2019). DOI: 10.35940/ijrte.D5405.118419.
- [16] Viola, P., Jones, M.J.: Robust real-time face detection. *International Journal of Computer Vision* 57(2), 137–154 (2004). DOI: 10.1023/B:VISI.0000013087.49260.fb.
- [17] Babitha D., Jayasankar T., Sriram V.P., Sudhakar S., Prakash K.B., Speech emotion recognition using state-of-art learning algorithms 2020, *International Journal of Advanced Trends in Computer Science and Engineering*, 9(2), 1340–1345, DOI: 10.30534/ijatcse/2020/67922020.
- [18] Bharadwaj Y.S.S., Rajaram P., Sriram V.P., Sudhakar S., Prakash K.B. Effective handwritten digit recognition using deep convolution neural network, 2020, *International Journal of Advanced Trends in Computer Science and Engineering*, 9(2), 1335–1339, DOI: 10.30534/ijatcse/2020/66922020.

- [19] Kim, S. and Yoon, K. (2022). Deep and Lightweight Neural Network for Histopathological Image Classification. *Journal of Mobile Multimedia*, 18(06), 1913–1930. <https://doi.org/10.13052/jmm1550-4646.18619>.
- [20] Shevchuk, B., Ivakhiv, O., Zastavnyy, O., and Geraimchuk, M. (2023). Telemonitoring of Human Biomedical and Biomechanical Signals. *Journal of Mobile Multimedia*, 19(03), 877–896. <https://doi.org/10.13052/jmm1550-4646.19310>.

## Biographies



**Krishnakanth Medichalam** completed his B.Tech in Computer Science and Engineering with Specialization in Bioinformatics from Vellore Institute of Technology. He is currently working as UI Developer in Tata Consultancy Services. His current research interests are Image processing, Modern Web Development and Natural Language Processing.



**V. Vijayarajan** received his B.E., Computer Science and Engineering from Madras University in 2001, M.E., Computer Science and Engineering from Anna University in 2007, and Ph.D., in Computer Science and Engineering from Vellore Institute of Technology in 2016. He is currently working as Associate Professor in School of Computer Science and Engineering, VIT University, Vellore, India. He has published 45+ articles in peer reviewed

Journals which include reputed Conferences. His current research interests are Image Retrieval, Ontological Engineering, Operating Systems, Formal Languages and Automata Theory, Wireless Sensor Networks, Mobile Cloud Computing, Machine Learning and Deep Learning.



**V. Vinoth Kumar** is an Associate Professor at School of Information Technology and Engineering, VIT University, Tamilnadu, India. He is currently being as an Adjunct Professor with School of Computer Science, Taylor's University, Malaysia. He is having more than 12 years of teaching experience. He has organized various international conferences and delivered keynote addresses. He has published more than 60 research papers in various peer-reviewed journals and conferences. He is a Life Member of ISTE and a professional member of IEEE. He is the Associate Editor of International Journal of e-Collaboration (IJeC), International Journal of Pervasive Computing and Communications (IJPCC) and Editorial member of various journals. His current research interests include Wireless Networks, Internet of Things, machine learning and Big Data Applications. He has filled 9 patents in various fields of research.



**I. Manimozhi Iyer**, currently working as an Associate Professor in the Department of Computer Science and Engineering, East Point College of

Engineering and Technology, India. She has published more than 25 research articles in various journals. Her research interest in the fields of Algorithms, Artificial Intelligence and Artificial Neural networks.



**Yaswanth Kumar Vanukuri** completed his B.Tech in Computer Science and Engineering from Vellore Institute of Technology. He is currently working as Sr.Tech Associate in Bank of America. His current research interests are Image processing and Machine Learning.



**V. B. Surya Prasath**, is a mathematician with expertise in the application areas of image processing, computer vision and machine learning. He received his PhD in mathematics from the Indian Institute of Technology Madras, India in 2009. He has been a postdoctoral fellow at the Department of Mathematics, University of Coimbra, Portugal, for two years from 2010 to 2011. From 2012 to 2015 he was with the Computational Imaging and VisAnalysis (CIVA) Lab at the University of Missouri, USA as a postdoctoral fellow, and from 2016 to 2017 as an assistant research professor. He is currently an assistant professor in the Division of Biomedical Informatics at the Cincinnati Children's Hospital Medical Center, and at the Departments of Biomedical Informatics, Electrical Engineering and Computer Science,

University of Cincinnati from 2018. He had summer fellowships/visits at Kitware Inc. NY, USA, The Fields Institute, Canada, and IPAM, University of California Los Angeles (UCLA), USA. His main research interests include nonlinear PDEs, regularization methods, inverse and ill-posed problems, variational, PDE based image processing, and computer vision with applications in remote sensing, biomedical imaging domains. His current research focuses are in data science, and bioimage informatics with machine learning techniques.



**B. Swapna**, is currently working as Assistant Professor, ECE Department in Dr. M.G.R. Educational and Research Institute (Deemed to be University) Chennai, India. She has completed her ph.D in Internet of things, Master's degree with honors in Applied Electronics secured Anna University 14th rank from SBC Engineering College, Arani and Bachelor's degree in Electronics and Communication Engineering from from SBC Engineering College, Arani, India. Her areas of interest include VLSI, Embedded systems, IoT and Electronic circuits.

