International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:8, No:11, 2014

Driver Fatigue State Recognition with Pixel Based Caveat Scheme Using Eye-Tracking

K. Thulasimani, K. G. Srinivasagan

Abstract**—**Driver fatigue is an important factor in the increasing number of road accidents. Dynamic template matching method was proposed to address the problem of real-time driver fatigue detection system based on eye-tracking. An effective vision based approach was used to analyze the driver's eye state to detect fatigue. The driver fatigue system consists of Face detection, Eye detection, Eye tracking, and Fatigue detection. Initially frames are captured from a color video in a car dashboard and transformed from RGB into YC_bC_r color space to detect the driver's face. Canny edge operator was used to estimating the eye region and the locations of eyes are extracted. The extracted eyes were considered as a template matching for eye tracking. Edge Map Overlapping (EMO) and Edge Pixel Count (EPC) matching function were used for eye tracking which is used to improve the matching accuracy. The pixel of eyeball was tracked from the eye regions which are used to determine the fatigue state of the driver.

Keywords**—**Driver fatigue detection, Driving safety, Eye tracking, Intelligent transportation system, Template matching.

I. Introduction

THE increasing number of transportation accidents has become a serious problem for society. According to [12], driver fatigue has been one of the major causes of traffic accidents all over the world and mostly in India. As per National Highway Traffic Safety Administration (NHTSA) survey, there are approximately 56,000 sleep-related road crashes annually which results in 40,000 injuries and 1,550 fatalities. The traffic accidents will be largely decreased if a judging rule is found to determine whether drivers stay awake or not. A warning alarm was made when the drivers begin to fall sleeping. A research is made in fatigue detection algorithm which is also a key technology for smart vehicles driving. The driver fatigue problem has become an important factor for causing traffic accidents. Driver fatigue is a major source of car accidents, as sleepy drivers are not capable of making rapid decisions. Many governments started education program to alert people to the dangers of driving while tired. Recently many safety systems are followed to avoid transportation accidents. Passive safety system like airbags, crashworthy body structures and seat belts help to reduce the effects of an accident. In contrast, active safety systems help drivers avoid accidents by monitoring the state of the vehicle [2], the driver, or the surrounding traffic environment and alert the driver or control interventions.

In this paper, the driver fatigue detection techniques can be divided into three generic sections. First the systems that investigate the physiological information of drivers [1], [4], [11] focus on measuring the physiological changes of drivers. The fatigue and drowsiness of the car driver were determined accurately using the physiological features. A significant struggle was made to measure the driver's physiology in the laboratory. Electroencephalogram (EEG), Electrocardiogram (ECG) and Electromyography (EMG) are the common general methods used for measuring the physiological changes of drivers. EEG is useful for determining the presence of on-going brain activity which is the reference point for calibrating the other measures of sleep and fatigue [14]. Next, find the vehicles behaviour of the system to detect the driver drowsiness such as steering wheel movements, driver's grip force on the steering wheel [3], speed, acceleration, changing course, turning angle, lateral position, gear changing, and braking etc. Some of the methods can be used for detecting the driver's physical changes during fatigue and observe the physical changes in facial features [2], [5]. Last using a web camera to detect the physical responses in eyes [6], [7], [9] [10], [13], [15] which indicated the responses of eyes have high relativity with driver fatigue or not. The combination of fatigue state and tracking of eye during driving detection system returns the best solution.

II. Methodology

The proposed methodology is based on eye tracking in real-time driver fatigue detection for the effective vehicle control system. This system detects the driver fatigue state based on eye tracking which comes under an active safety system. An ordinary color webcam was used to capture the videos of the driver for fatigue detection. The captured video was converted into number of frames, in which the first frame is used for initial face detection and eye location. If the driver face and eye were not properly detected in the first frame, the sequence of frames can be processed to detect the driver's face and eyes. The current eye images were used as the dynamic templates for eye tracking on subsequent frames. Based on the eye position [10], the driver fatigue state is detected. If the driver's eye cannot be detected properly, the face and eye location were detected on the current frame. For the illumination change problem, YC_bC_r color model was adopted to represent the skin region. Since eyes have the nature of sophisticated edges, Canny edge operator and the projection technique was used to locate eyes positions. The eyes images are used as the templates for eye tracking in subsequent frames. The tracked

K. Thulasimani is an Assistant Professor at Government College of Engineering, Tirunelveli, Tamilnadu, India (e-mail: thulasimani@gcetly.ac.in).

Dr. K. G. Srinivasagan is Professor at National Engineering College, Kovilpatti, Tamil Nadu, India (e-mail: drkgcse@ nec.edu.in).

eyes images were used to determine whether the eyes are open or closed. Fig. 1 represents the system flow diagram for driver fatigue detection system.

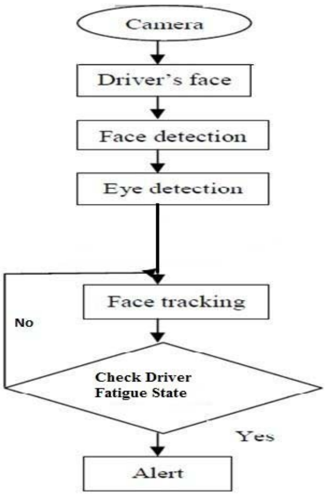


Fig. 1 System flow diagram for driver fatigue detection system

A. Face Detection

The captured color video was converted into number of frames. The RGB color space representation is used to convert the color frames into HSI color space for face detection. The HSI color space is used to eliminate the brightness factor from affecting the skin color detection to handle with the illumination change problem. A suitable range of threshold HUE values can be defined to detect the face region correctly as well as horizontal and vertical projections of the face region can also be detected. In a frame, the upper 2/5 of the detected face region was called the *eye region* was used to detect the location of the eye. Fig. 2 represents some of the samples frames for driver fresh and tiredness of the captured video.

Fig. 2 Sample input frames for driver fatigue detection system

B. Skin Segmentation

Using skin segmentation techniques, the human face was extracted from the video frame. Digital images are usually adopted to represent color in RGB color space. A number of colors in the RGB space was not only displays the hue value but also contains the brightness of the image. Two colors having the same hue value having different intensities, it would be viewed as two different colors by the human optical system. The brightness factor is excluded from colors. The skin pixel and non-skin pixels can accurately distinguished which is not affected by shadows or light changes of the skin. Using YC_bC_r color space, the chromatic red and chromatic blue values are largely independent of skin color or lighting conditions. Equation (1) represents to convert between RGB color space into YC_bC_r .

$$\left.\begin{aligned} Y &= 0.213R + 0.715G + 0.072B \\ C_b &= -0.117R - 0.394G + 0.511B + 128 \\ C_r &= 0.511R - 0.464G - 0.047B + 128 \end{aligned}\right\} \quad (1)$$

where Y is the luminance component, C_b and C_r are blue and red difference of chroma components. In YC_bC_r space, Y component was not changed between various skin color. The C_b and C_r values are used to detect the pixels on the skin images. For representing the skin color, the threshold range value of C_b is 80 $\leq$ C_b $\leq$ 120 and C_r value is 133 $\leq$ C_r $\leq$ 173. For each pixel verify the threshold value of C_b and C_r, which was above the skin location treated as black and the pixel value changes to 1 or it was white, the pixel value changes to 0. The result of the skin location technique is converted into white and the background areas of face region around the driver face. Fig. 3 represents the face region was detected by finding the biggest white connected component and segmenting the area from the frame.

Fig. 3 Extract skin region from face detection

C. Skin Color Model

A color model is an abstract mathematical model describing the way of colors which is represented as tuples of numbers. RGB is a color space or model which represent each pixel of three values such as red, green, blue and each value vary from 0 to 255. All three RGB components need to be of equal bandwidth to generate any color within the RGB color cube. YC_bC_r is a color space that represents any color in three values, C_b and C_r are the blue-difference and red- difference of chromatic components and Y is the luminance component. Y was defined to have a nominal 8 -bit range of 16 to 235. C_b and C_r were defined to have a nominal range of 16-240. For every two horizontal Y samples, there is one C_b and C_r sample. Each sample is normally 8 bits or 10 bits per component. Equation (2) represents to convert 8 bit RGB data into 16 - 256 bit nominal range of YC_bC_r

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112 \\ 112 & -93.786 & -18.214 \end{bmatrix} * \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2)$$

The upper half of the detected face region is called the eye region which is used for eye location. The original color information of the detected eye region is first converted into grayscale. Gray scale image provides less information about the each pixel. Grayscale image has all equal intensity value of red, blue and green components in RGB space. It is required to specify a single intensity value for each pixel. Color image of each pixel has a different intensity value of RGB component. The Canny edge operator is used for edge detection in the gray-level eye region.

D.Edge Detection

Edge detection is a set of mathematical methods which identifies points in a digital image at which the image brightness changes abruptly has discontinuities. The image brightness is normally organized into a set of curved line segments which is called edges. Canny Edge Detection Algorithm consists of five separate steps such as smoothing, finding gradients, non-maximum suppression, double thresholding and edge tracking by hysteresis.

The images taken from a camera contains some amount of noise. Smoothing is the process of removing the noise from the blurred image. The noise of the image was reduced by using Gaussian filter and the images of incorrect edges were smoothened. Equation (3) represents the standard derivation $\sigma = 1.4$ of Gaussian filter kernel function.

$$B = \frac{1}{59} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} \quad (3)$$

Canny Edge detection algorithm was used to mark the edges of grayscale image where the gradients of the image have high value of magnitudes. The Sobel operator [8] was used the find the gradients value of each pixel in a smoothened image. The gradient magnitudes also known as the edge strengths can be determined as a Euclidean distance measure by applying the law of Pythagoras as shown in (4) and (5):

$$|G| = \sqrt{G_x^2 + G_y^2} \quad (4)$$

$$|G| = |G_x| + |G_y| \quad (5)$$

where G_x and G_y are the gradient value in the x- and y-directions respectively. The gradient magnitudes value of an image always represents relatively smooth. The broad edges are not clearly indicated where the edges are present in an image. Equation (6) represents to determine the direction of the broad edges.

$$e = \arctan\left(|G_y| / |G_x|\right) \quad (6)$$

Non-maximum suppression is marked the local maxima as edges. The gradient magnitude of the blurred edges is converted into a sharp edge in which all the local maxima value present in the gradient image is preserved and the maximum value was suppressed. The non-maximum suppression algorithm was applied to every pixel in the gradient image. (i). the gradient direction θ is round up to the nearest 45°. (ii). All the edge strength of the current pixel are compared with edge strength of the pixel present in the Positive gradient & negative gradient direction. (iii). the edge strength of the current pixel value is greater than the preserve value of the edge strength otherwise suppress it.

Double threshold method was used to find the potential edges of the image by its threshold value. After the non-maximum suppression step, the edge pixels remain and marked with the strength pixel by pixel. Many edges are true edges in the image, but some are caused due to noise or color variation for an instance due to the rough surfaces. The easy way would be to utilize a threshold, to find the only edges stronger having appropriate value would be preserved. The Canny edge detector algorithm uses double thresholding mechanism. Edge pixels having strong threshold are marked as strong; edge pixels having weaker than the low threshold are suppressed and edge pixels between the two thresholds value are marked as weak.

Edge tracking is the final edges are determined by combining all the edges that are not connected. Strong edges are considered as certain edges, & immediately included in the final edge image. Weak edges are considered when they are connected to strong edges. The other logic such as noise & other small variations are unlikely to give the result in a strong edge. Thus strong edges are present due to the true edges in the original image. Weak edges can be present due to true edges or noise or color variations. It uniformly distributed on the entire image and only a small amount will be located at the strong edges. Due to true edges the weak edges are connected directly to the strong edges. Fig. 4 represents the edge detection of the eye region using canny edge detection algorithm.

Fig 4 Edge detection from the Eye Region

E. Eye Tracking

Consider an eye template g_t of width w and height h, located at the position (a,b) in the original frame. The search area of a new frame for eye tracking is the eye template position by expanding a reasonable number of pixels in each of four directions: left, right, up, and down is shown in Fig. 5. Let d_{x_max} and d_{y_max} be the maximum displacements of the x - axis and y - axis, respectively. Thus, the size of the search area is $(w+2d_{x\_max})\times(h+2d_{y\_max})$, and the number of *search points* for the exhaustive search is equal to $(2d_{x\_max}+1)\times(2d_{y\_max}+1)$. This search area in the new color frame is first converted into a gray-level one for eye tracking. Equation (7) refers the Mean Absolute Difference (MAD) matching function is used for eye template matching:

$$M(p,q)=\sum_{x=0}^{w-1}\sum_{y=0}^{h-1}\left|g_t(x,y)-g_t(x+p,y+q)\right| \tag{7}$$

where p and q are displacements of the x_{axis} and y_{axis} respectively in which $(a-d_{x\_max})\le p\le(a+d_{x\_max})$ and $(b-d_{y\_max})\le q\le(b+d_{y\_max})$. If $M(p^*,q^*)$ is the minimum value within the search area, point (p^*,q^*) is the most matching position of g_t, and let f_e denote the matched eye sub-image in the current color frame for fatigue detection. Finally, update the position (a,b) of g_t to be the new position (p^*,q^*) for tracking on subsequent frames. Point (p^*,q^*) is the most matching position of g_t.

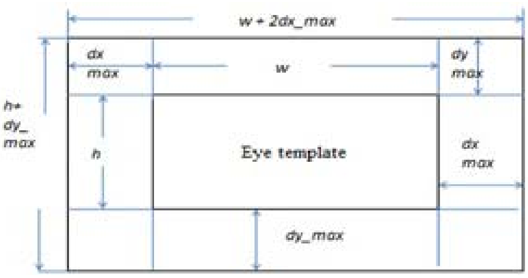
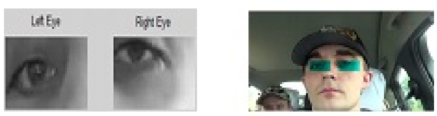


Fig. 5 Search area of an eye template for tracking

This search area in the new color frame is first converted into a gray-level one for eye tracking. Fig. 6 represents the eye template and the position of driver's eye are tracked using edge map overlapping and edge pixel count matching function.



(a). Eye Template (b). Position of Eye Templates in the image

Fig. 6 Eye template for tracking

F. Template Matching

Template Matching is a technique for finding small parts of an image which matches a template image. It uses a convolution mask template to detect a personalized image to a specific feature of the search image. This technique can be performed on gray images or edge images. Where the image structure matches the mask structure, the output of the convolution will be highest at places and where large image values get multiplied by large mask values. This method is usually implemented by picking out a part of the search image to use as a template. Then the center of the template $T(x_t,y_t)$ is moved over each (x,y) point in the search image. The sum of products is found between $S(x,y)$ and $T(x_t,y_t)$ over the whole area spanned by the template. The position with the highest score is the best position of the template with respect to the search image is considered.

G. Edge Map Overlapping Matching Function

In a human face, there are eyelashes, eyelids, irises, and pupils around the eyes. Therefore, the eyes have prominent edges around them. This complicated edge feature of the eye has been used for eye location. However, it can also be further used for eye tracking. In order to use edge feature for designing new matching functions, the search area, and the eye template must first be converted into their corresponding edge maps. Let $\boldsymbol{e_s}$ and $\boldsymbol{e_t}$ be the edge maps of the search area and the eye template respectively. Equation (8) represents the Edge Map Overlapping matching function is defined as:

$$E(p,q)=\sum_{x=0}^{w-1}\sum_{y=0}^{h=1}e_t(x,y).e_s(x+p,y+q) \tag{8}$$

where the operator "·" represents the logical-AND operation p and q are displacements of the x_{axis} and y_{axis} respectively. The summation is taken over the region where $\boldsymbol{e_s}$ and $\boldsymbol{e_t}$ overlap.

H. Edge Pixel Count Matching Function

Since the eyes have prominent edges around them within the search area, the number of edge pixels could be another stable feature for eye tracking. In this case, the eye template is regarded as a sliding window moving within the search area and only need to count the number of edge pixels in the moving window. Equation (9) represents the Edge Pixel Count matching function is defined as:

$$C(p,q)=\sum_{x=0}^{w-1}\sum_{y=0}^{h-1}e_s(x+p,y+q) \tag{9}$$

where p and q are displacements of the x_{axis} and y_{axis} respectively from which $(a-d_{x\_max})\le p\le(a+d_{x\_max})$ and $(b-d_{y\_max})\le q\le(b+d_{y\_max})$.

I. Fatigue Detection

Once the face of the driver is detected, the eye of the driver is successfully tracked by continuously monitoring the variations of the eye. The feature of the darker color eyeballs in the eye templates are used directly for fatigue detection. The property of the eyeball colors much darker is a quite stable feature. The eye templates are converted to the grayscale model and which are inverted (negated). The original darker eyeballs become brighter ones in the inverted image and the pixels with low saturation values are regarded as eyeball pixels. According to the observation, the saturation values of eyeball pixels normally ranges between 0.00 and 0.14. This observation is used to distinguish whether a pixel in an eye template is viewed as an eyeball pixel. Fig. 7 represents that when the eyes are open, there are some eyeball pixels and when the eyes are closed, there are no eyeball pixels present. By checking the eyeball pixels, it is easy to detect whether the eyes are open or in the fatigue state.

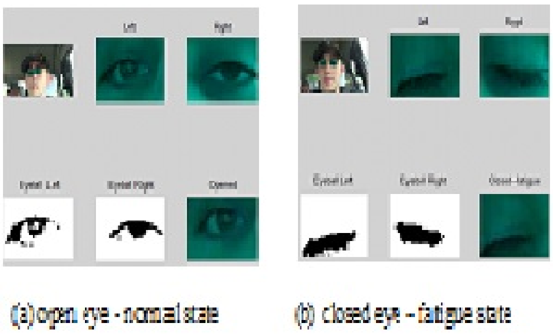


Fig. 7 Results of eyeball detection

III. Experimental Results and Analysis

The testing environment of the driver fatigue detection system was implemented using MATLAB with a personal computer system with 2GB RAM. The duration of a normal eye blink (the small peak) is usually less than 3 frames at the capture rate of 30 frames per second. If the duration for an eye-close is more than 5 frames, the distance of the car will move more than 5.5 meters at the speed of 80 kilometers per hour in a highway. It is very dangerous for the driver to ignore the status of the road when driving in such a condition. For this reason, the proposed system is defined that when the eyes close over 5 consecutive frames the driver is regarded as dozing off. This parameter can be adjusted by users according to their needs. Table I shows the experimental results of the driver fatigue detection system on a sample of test videos. In these experiments, the driver is regarded as dozing off when their eyes close over 5 consecutive frames. In this table, each field is stated below:

$$CorrectRate(CR) = \frac{\text{Number of frames of correct dozing}}{\text{Number of frames of real dozing}} \quad (8)$$

$$\Pr ecissionRate(PR) = \frac{\text{Number of frames of correct dozing}}{\text{Number of frames of detected dozing}} \quad (9)$$

It was shown that the system could reach 100.0% correct rate of fatigue detection while the precision rate could still achieve 89.3%.

TABLE I
Results of Fatigue Detection

Video	No. Of Frame	No. Of Real Dozing Frame	No. Of Detected Dozing Frame	No. Of Correct Dozing Frame	Correct Rate	Precision Rate
1	433	2	2	2	100	100
2	1524	4	4	4	100	100
3	2634	3	3	3	100	100
4	2717	15	18	15	100	83.3
5	1399	1	1	1	100	100
Total	8707	25	28	25	100	89.3

IV. Conclusion

The proposed method is a non-invasive system to localize the eyes and monitor fatigue was developed. The captured video frames are converted into YC_bC_r color space from which the drivers' face was detected and cropped from a frame. The eyes in the face region are estimated and extracted using canny edge detector. The matching functions such as Edge Map Overlapping and Edge Pixel Count are used specifically for eye tracking to improve matching accuracy. This system will monitor the eye movement to detect the fatigue state of the driver and a warning signal is given to alert the driver. It achieves high accuracy and reliable detection of drowsiness. It also offers a non-invasive approach to detecting the drowsiness without the annoyance and interference. Since a large number of road accidents occur due to the driver drowsiness, thus this system will be very helpful in preventing many accidents.

References

[1] T. C. Chieh, M. M. Mustafa, A. Hussain, E. Zahedi, B. Y. Majlis, "Driver Fatigue Detection Using Steering Grip Force", *Proc. IEEE Student Conference on Research and Development*, Putrajaya, Malaysia, 2003, pp.45-48.

[2] K. J. Cho, B. Roy, S. Mascaro, and H. H. Asada, "A Vast DOF Robotic Car Seat Using SMA Actuators with a Matrix Drive System," *Proc. IEEE Robotics and Automation*, New Orleans, LA, USA, Vol.4, 2004, pp.3647- 3652.

[3] R. C. Coetzer and G. P. Hancke, "Eye Detection for a Real-Time Vehicle Driver Fatigue Monitoring System," Proc. *2011 IEEE Intelligent Vehicles Symposium, Baden-Baden, Germany*, 2011, pp. 66-71.

[4] W. Dong and X. Wu, "Driver Fatigue Detection Based on the Distance of eyelid," *Proc. IEEE VLSI Design and Video Technology*, Suzhou, China, 2005, pp. 365-368.

[5] H. Gu, Q. Ji, and Z. Zhu, "Active Facial Tracking for Fatigue Detection", *Proc. 6th IEEE Workshop on Applications of Computer Vision*, Orlando, FL, USA, 2002, pp. 137-142.

[6] H. Gu and Q. Ji, "An Automated Face Reader for Fatigue Detection," *Proc. 6th IEEE International Conference on Automatic Face and Gesture Recognition*, Seoul, Korea, 2004, pp.111-116.

[7] W. B. Horng and C. Y. Chen, "A Real-Time Driver Fatigue Detection System Based on Eye Tracking and Dynamic Template Matching." *Tamkang Journal of Science and Engineering*, Vol.11, No.1, 2008, pp.65-72.

[8] R.C. Gonzalez and R.E. Woods, *Digital Image Processing*, *Second Edition*, Prentice Hall, Upper Saddle River, NJ, USA, 2002.

[9] T. Ito, S. Mita, K. Kozuka, T. Nakano, and S.Yamamoto, "Driver Blink Measurement by the Motion Picture Processing and Its Application to

Drowsiness Detection," *Proc. IEEE 5th International Conference on Intelligent Transportation Systems*, Singapore, 2002, pp. 168-173.
[10] Q. Ji, Z. Zhu, and P. Lan, "Real-Time Nonintrusive Monitoring and Prediction of Driver Fatigue," *IEEE Transactions on Vehicular Technology*, Vol.53, No.4, 2004, pp.1052-1068.
[11] M. A. Recarte and L. M. Nunes, "Effects of Verbal and Spatial-Imagery Tasks on Eye Fixations while Driving," *Journal of Experimental Psychology: Applied*, Vol.6, No.1, 2000, pp.31-43.
[12] Smart Motorist, Inc., "Driver Fatigue is an Important Cause of Road Crashes", *http://www.smartmotorist.com/traffic-and-safety-guideline//driver-fatigue-is-an-important-cause-of-road crashes.html.*
[13] H. Wang, L. B. Zhou, and Y. Ying, "A Novel Approach for Real Time Eye State Detection in Fatigue Awareness System," *Proc. 2010 IEEE International Conference on Robotics Automation and Mechatronics*, 2010, Singapore, pp. 528-532.
[14] J. H. Yang, Z. H. Mao, L. Tijerina, T. Pilutti, J. F. Coughlin, and E. Feron, "Detection of Driver Fatigue Caused by Sleep Deprivation," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, Vol. 39, No. 4, 2009, pp. 694-705.
[15] K. P. Yao, W. H. Lin, C. Y. Fang, J. M. Wang, S.L. Chang, and S. W. Chen, "Real-Time Vision-Based Driver Drowsiness/Fatigue Detection system, *Proc. IEEE 71st Vehicular Technology Conference*, Taipei, Taiwan, 2010, pp. 1-5.

K. Thulasimani was born on 05/04/1976. She obtained her B.E. Computer Science and Engineering from Government College of Engineering, Tirunelveli, Tamilnadu, India. She has done M.E Computer Science Engineering at Anna University, Chennai, Tamilnadu, India, currently pursuing Ph.D. in the field of video processing and analysis.

She is working as Assistant Professor in Department of Computer Science and Engineering, Government College of Engineering, Tirunelveli, Tamilnadu, India. Her research interests are Object identification and tracking, Object recognition, Video surveillance and Computer Vision.

She is a Life Member of ISTE and IE (I) chapters.

Dr. K.G. Srinivasagan was born on 10/07/1967 in Kovilpatti. He graduated Bachelors degree from Madurai Kamaraj University, Madurai, Tamil Nadu, India in Computer Science and Engineering in the year 1989. He received his master's degree in Computer Science and Engineering from Bharatiyar University, Coimbatore, Tamil Nadu, India in the year 1998. He received his Ph.D degree in Image Processing from Manonmaniam Sundaranar University, Tirunelveli, Tamilnadu, India in the year 2009.

He is working as a Professor in Department of Computer Science and Engineering, National Engineering College, Kovilpatti, Tamilnadu, India He has published Papers in Reputed Journals and his research interest includes Image processing, Data mining, Natural Language Processing, Cloud Computing.

He is a life member of IEI, CSI, and ISTE chapter.