Journal of Excellence in Computer Science and Engineering, Vol. 2(2) 2016, pp. 1-9

RESEARCH ARTICLE

Foreground Segmentation for Live Videos by Texture Features

*M.R Resmi[1], E Arun[2]

[1]P.G Scholar, Department of Computer Science and Engineering, T.K.M Institute of Technology, Kerala, India.

[2]Professor, Department of Computer Science and Engineering, Marian Engineering College, Kerala, India.

Received-12 March 2016, Revised-9 May 2016, Accepted-30 July 2016, Published-23 August 2016

ABSTRACT

This paper presents a method to extract the foreground images from live videos by means of automatic object segmentation. The parameters such as colour, motion of the pixel and image texture or more specifically texture constraints are used for segmentation. A cellular neural network which combines both colour as well as motion of pixels which varies from frame to frame is implemented which helps in accurately separating the boundaries and thus reducing misclassifications. The global motion of pixels is calculated by computing the forward and backward displaced frame differences (DFDs) with the respect to the current frame. The texture constraint for each pixel to be labeled is calculated from the difference between their corresponding texture descriptors and the texture prior models which is provided by the Local Binary Pattern (LBP) histogram. Finally by means of the randomized texton searching algorithm and graph cut frame work the foreground is extracted from the video.

Keywords: Cellular neural network, DFDs, Local binary pattern histogram, Texture, Texture prior models, Texture descriptors.

1. INTRODUCTION

Image segmentation is of much importance in video analysis as it helps in separating images which in turn can be used for various applications viz surveillance systems, teleconferencing and video editing. This separation can be either within the image or between images depending on the case under study. Such kind of segmentation is mainly base on low level cues such as homogeneity, intensity, contours etc., The techniques mostly follow a uniform procedure in which the backgrounds are modelled first. Next any changes which occur in the incoming video frames are monitored. Multiple difficulties arise while modelling the background. For example, there may be shapes and structures available in the background. Moreover the background objects may change in colour and intensity from frame to frame. In addition to it, the videos might have been taken under different lighting scenarios which make the process more complex. Hence a good foreground segmentation technique should be able to

1)Develop a good background model and

2)Should be robust to all kinds of changes taking place between frames.

The background modelling methods can be broadly classified into three categories. They are pixel based, region based and hybrid based methods. In pixel based methods at least one traversal through the entire pixels is performed which makes it an efficient technique. The histograms of each pixel are calculated and used for the segmentation process. In region based methods the attributes of the pixels viz colour, intensity etc., are taken into account. Hybrid based methods combine the pixel base and region based methods for producing sophisticated results. Background

*Corresponding author. Tel.: +919443595337
Email address: mrresmimr@gmail.com (M.R.Resmi)

Double blind peer review under responsibility of DJ Publications
http://dx.doi.org/10.18831/djcse.in/2016021001

modelling methods can also be categorized as parametric and non-parametric methods. The foreground image segmentation program has many applications, one of them being video conferencing. If the background and foreground images are separated, the background can be replaced by another background which may help in improving the artistic beauty of the video. The extracted foreground objects are more suitable for transmission. Using modern editing tools the extracted images can be combined to produce new results.

We emphasize on foreground segmentation in which objects or more specifically foreground objects are extracted from real-time videos.

2. LITERATURE REVIEW

[1] proposed that the background at each pixel can be easily modelled using a Gaussian distribution function. However during iterations, the updating capability of the model was found to be much low. For videos and frames which undergoes dynamic changes this is undesirable. Hence [2] proposed a new method which is called the Gaussian Mixture Model (GMM). Here each pixel was modelled with a mixture of K gaussian functions. However in this method the initialization of parameters consumed a lot of time. [3] proposed a threshold decision method for separating the foreground objects. Camera noise was assumed to be the only factor which affects the threshold. However this was not the case when the experiment was carried out in the real time. There were other factors too which affected the threshold and hence this method was limited to particular cases. A non-parametric algorithm named Self Organizing Background Substraction (SOBS) was proposed by [4] based on artificial neural networks. Again time constraint was found to be a major problem as SOBS was interested in measuring the sample frame between weight vectors. [5] proposed a code book based foreground segmentation. Here for each pixel, based on training sequences, a code book is created which stores multiple code words. The main disadvantage of this method was that, if the foreground and background pixels were the same in colour, the foreground was incorrectly segmented. [6] proposed a sample based consensus method SACON to effectively segment the foreground and background models. In this method, for each pixel at a time t, a cache of N background samples is made. If for every pixel, the new value matches with most of the values in the cache then it was classified as background. But this issue fails with a first in first out scheme. A non-parametric algorithm named Vibe was proposed by [7]. In this technique the first frame of the video was used to initialize the background model based on the assumption that the neighbouring pixels share the similar temporal distribution. Even though Vibe is superior to many state of the art technologies, one of the profound disadvantage was that in the case of dynamic video backgrounds it requires manual adjustment to adapt to background changes. [8] combined Vibe and SACON to create a Pixel Based Adaptive Segmenter (PBAS) method. Here a history of N image values is taken as the background model. The updation was done similar to vibe. In PBAS the parameters are taken as adaptive state variables. In the case of static backgrounds the method did not met the expected levels. [9] proposed a non-parametric background modelling method called Kernel Density Estimator (KDE). In this method the probability of the intensity of each pixel is estimated at time t. If the probability was less than the threshold value, then it will be treated as foreground and otherwise as background. The threshold is adjustable over a long range of values. KDE has the disadvantage that in the case of large frames, it has to keep the N frames in memory and is very time consuming. [10] proposed a heuristic matching algorithm to separate the foreground and background images. Here each incoming frames are compared with a fixed data set and the images are segmented into foreground and back ground images. [11] used a vertical edge detection scheme to extract the numbers alone from the license plates. Here numbers were treated as the foreground and the other regions were treated as background. In [12, 13, 14], for each pixel location, a one-class Support Vector Machine (SVM) was maintained. Based on the colour intensity each pixel was mapped on to the corresponding SVM and then the pixels were labelled jointly. Next a graph cut technique was used to segment the foreground image. The major disadvantage of this approach is that, here only the colour and alpha values were used for separating the background and foreground which is

inadequate for processing live videos. In some other related works, the first frame of the video was designated as f0. Canny edge detector was applied to this frame to find the boundaries between the foreground and background image. The foreground pixels were extracted by an iterative threshold method where a particular value is kept as the threshold. If the value of the pixel is above the threshold it will be categorized as foreground and if not as background. Hence we get a group of foreground pixels in the initial frame. Next a cellular neural network was used to extract the foreground of the incoming frames.

By considering all the above mentioned literatures in our approach, we used a region based method to extract the foreground image. The region based method is used as it is very efficient in noisy images particularly in detecting the borders. The primary idea is to find the homogenous regions in an image and segment the image based on it. The homogenous regions are those which share some similar property. We replace the C1SVMs used in [12] by the Cellular Neural Networks (CNN) along with randomized texton search algorithm. The CNN is used exclusively as it supports backward error propagation and contains a number of hidden layers. The segmentation errors are reduced with the increase in hidden layers. Colour, motion of the pixels and image texture of the video frames are referenced as the homogenous parameters. The proposed idea efficiently processes the live video scene by automatically segmenting the object (foreground).

3. SYSTEM MODEL

An overview of the system model is explained as follows. The input to the proposed system is the input video frame. First the colour and pixel motion constraints are found out and are trained using a cellular neural network for distinguishing the boundaries and to avoid misclassifications. Next the texture prior models of the foreground and background are established. The Local Binary Pattern (LBP) histogram helps in providing the texture prior model for establishing texture constraints. Next an energy function comprising of colour, motion of pixels and texture terms generated using a randomized texton searching algorithm is put forward. Then a graph-cut technique is employed to minimize the energy function resulting in the foreground.

3.1. Colour and motion of pixels

The colour constraint helps a pixel to assign itself, the label of the colour prior model which has a smaller distance to that pixel. In dealing with images having different colours in foreground/background, the colour term is capable of obtaining satisfactory results. Assuming the motion of pixel from previous frame Ft-1 to Ft is identical to its motion from Ft to Ft+1, global motion compensation aligns background pixels of the Ft-1 and Ft+1 with that in frame Ft to high accuracy, while motion compensated foreground pixels tend to exhibit high error. The magnitude of errors in frame Ft-1 and Ft+1 with those in Ft using the computed global motion are termed as previous (backward) and next (forward) displaced frame difference (DFDs) respectively.

3.2. Cellular Neural Network (CNN)

The dynamic equation of a CNN may be approximated by the following difference equation (3.1)

$$\frac{du_{ik}}{dt} = \frac{1}{C}\left[X_{ik} - \frac{u_{ik}}{R} + \sum_{i\in I}\sum_{j\in N(i)} \mathrm{r}_{ij}^{kl}\, g\left(u_{jl}\right)\right] \quad (3.1)$$

where R and C are the resistance and capacitance of the CNN, g(.) is a sigmoid function, X_{ik} and r_{ij}^{kl} are the constraints found and u_{jl} is the state of neuron promoting label l for pixel j.

3.3. Texture

Texture is the set of matrices calculated in image processing designed to quantify the perceived texture of an image. Texture gives the information about the spatial arrangement of colour or intensities in an image.

We define $E_1(l_i)$ as shown in equation (3.2)

$$\begin{cases} \mathrm{E}_1(\mathrm{l}_\mathrm{i} = 0) = \mathrm{g}_0\left(\mathrm{TD}_\mathrm{i}^\mathrm{F}, \mathrm{TD}_\mathrm{i}^\mathrm{B}\right), \\ \mathrm{E}_1(\mathrm{l}_\mathrm{i} = 0) = \mathrm{g}_0\left(\mathrm{TD}_\mathrm{i}^\mathrm{F}, \mathrm{TD}_\mathrm{i}^\mathrm{B}\right). \end{cases} \forall_\mathrm{i} \in \mathrm{U}_\mathrm{T} \quad (3.2)$$

where $E_1(l_i)$ is a texture constraint term measuring the cost of assigning a background (*li* = 0) or foreground (*li* = 1) label to node *i*.
and

$$\begin{cases} g_0\left(TD_i^F, TD_i^B\right) = \frac{TD_i^B}{TD_i^F+TD_i^B}, \\ g_1\left(TD_i^F, TD_i^B\right) = \frac{TD_i^F}{TD_i^F+TD_i^B}. \end{cases} \qquad (3.3)$$

In equation (3.3) TD_i^B and TD_i^F are node *i*'s distance to background and foreground texture prior models. For the model building, here we use the k means algorithm to cluster the pixels based on the user stroke. Firstly we establish the texture prior models. Based on texture prior model we compute TD_i^F and TD_i^B.

The texture model for foreground and background has 64 clusters denoted as $\{T_k^F\}_{k=1...64}$ and $\{T_k^B\}_{k=1...64}$ respectively. These clusters are trained through the clustering texton of pixels for B using k-means algorithm. Here LBP histogram is used [7] for texture extraction. The LBP histogram gives the histogram of the LBP values of the pixel in the texton. Figure 1 describes the grid of the LBP histogram of the image.

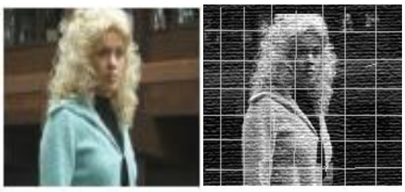

Figure 1 (a).Source image (b).Grid of LBP image

Firstly the image shown in figure 1(a) is divided into 64 blocks as shown in figure 1(b)

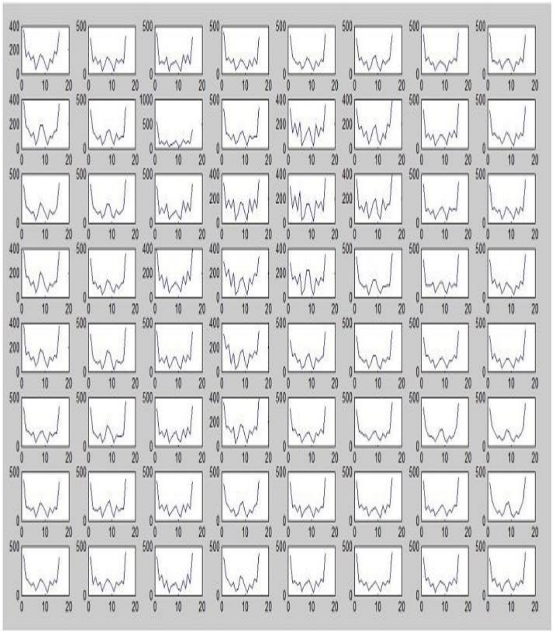

Figure 2.LBP histogram of each blocks in LBP image

Figure 2 gives the LBP histogram of each block of the image. The texture descriptor of pixels in foreground and background are needed to establish the texture prior model of foreground and background.

TD_i^F and TD_i^B are node i's distance to foreground and background texture prior models, denoted as $\{T_k^F\}_{k=1...64}$ and $\{T_k^B\}_{k=1...64}$ respectively. TD_k^F and TD_k^B are given by equation (3.4)

$$\begin{cases} TD_i^F = \min_{k=1....64} LH\_dist\left(T_i^F, T_k^F\right) \\ TD_i^B = \min_{k=1....64} LH\_dist\left(T_i^B, T_k^B\right) \end{cases} \qquad (3.4)$$

where T_k^B and T_k^F are the LBP histograms in k^{th} clusters of background and foreground respectively. T_i^B and T_i^F represent the LBP histogram denoting the textons of node i. T_i^B and T_i^F have the smallest distance (or most similar) to the foreground texture prior model and background texture prior model respectively.

$$\begin{cases} LH\_dist\left(T_i^F, T_k^F\right) = \sum_{j=1}^{n} \left|Bin_j\left(T_i^F\right) - Bin_j\left(T_k^F\right)\right| \\ LH\_dist\left(T_i^B, T_k^B\right) = \sum_{j=1}^{n} \left|Bin_j\left(T_i^B\right) - Bin_j\left(T_k^B\right)\right| \end{cases} \qquad (3.5)$$

In equation (3.5) n denotes the number of bins in the histogram in the experiment conducted, where n=16. $Bin_j(T_k^B)$ and $Bin_j(T_k^F)$ are the j^{th} bins of the local binary pattern histograms in the kth cluster of the background texture model and foreground texture model respectively.

To avoid the computing on texture less areas, the texture area in an image is found out using equation (3.6). T denote the threshold and is set to $T=\sqrt{500}$.$\forall i \in U_T$, $Var_{regional}(i) > T$.

$$Var_{regional}(i) \sqrt{\frac{\sum_{j=1}^{n}\left(G_j - AVG(G_{1...n})\right)^2}{n}} \qquad (3.6)$$

n is the number of pixels in a region. The 10X10 square window or region is assumed and $Var_{regional}(i)$ is the centre pixel i in the region. G_j denote the gray value of pixel j. $AVG(G_{1...n})$ denote the average grey value in the particular region.

3.4. Randomized texton searching

Consider a pixel which is to be labelled. As the size and position of the correct texton are unknown it is difficult to evaluate the distances between the pixel and the foreground/background texture prior models.

Each pixel may have lots of different candidate textons, which lead to different labels. Figure 1 can be regarded as an example. Out of the 64 grids some of them comes under background and some of them comes under foreground. We consider the histogram corresponding to grid in row 3 column 5 as one foreground and row 7 column 3 as another foreground. Again we consider row 5 column 2 as one background and row 7 column 8 as another background. Their corresponding LBP histograms are also considered. For each pixel, the histogram is found out. Then it is compared with the two foreground and two background histograms mentioned above. The closest is labelled as the texton of the corresponding pixel. However there are chances that misclassification may occur. Inorder to avoid this, a foreground representative texton, which is the most similar block to foreground and a background representative texton, which is the most similar block to background is found out for each pixel. Then, the distance between the foreground representative texton and the foreground model, and the distance between the background representative texton and the background model is used to estimate the label of p. To find the two representative textons a randomized texton searching algorithm is used. The LBP histogram of foreground/ background representative textons of a pixel i is denoted as T_i^B/T_i^F where T_i^B and T_i^F are the background and foreground representative textons and has the smallest distance to the background texture prior model and foreground texture prior model than the other block converging i, respectively. These two representative textons are obtained by randomized sampling:

(a) A set of blocks covering the current pixel is iteratively chosen by randomly sampling in size and centre position within a certain range.

(b) The distances between every block, which is described as an LBP histogram, and foreground/ background are computed. The block with the minimum distance is chosen as the foreground/background representative texton of the pixel.

The LBP histograms of the foreground/background representative textons for pixel i are denoted as TB_i and TF_i, respectively. Pixel i would be labeled as foreground if the distance between *TF*i and the foreground prior model is smaller than that between *TB*i and the background prior model and vice versa.

3.5. Graph-cut technique

Segmentation of the foreground object from the background will be formulated as a binary labelling problem. Given a set of sites S and set of labels L, the labelling problem will assign a label $F_p \epsilon$ L to each of the sites pϵS. The graph cut frame work can solve the labelling problem by using two labels. L= {0, 1} is the label set, where 1 corresponds to the foreground and 0 corresponds to the background. The energy function is formulated as in equation (3.7),

$$E_{(f)} = \sum_{\{p,q\}\in N} w_{pq}.T\left(f_p \neq f_q\right) \qquad (3.7)$$

The first term is the data term, which consists of constraints from the observed data and measures how sites like the labels that f assigns to them. $D_p(F_p)$ denotes the negative log likelihoods of the constructed background or foreground models. The value of $T(F_p \neq f_q F_q)$ is 0 if $F_p=F_q$ and 1 otherwise. This model can be regarded as a piecewise constant model because it encourages labelling consisting of several regions where sites in the same region have the same labels. In image segmentation, we want the boundary to lie on the edges in the image.

$$w_{pq} = e^{-\frac{(Ip-Iq)^2}{2\delta^2}} \frac{1}{(distance(p,q))} \qquad (3.8)$$

In (3.8) I(q) and I(p) are the colour value of site q and p respectively and distance(p, q) is the distance between sites p and q. Parameter δ gives the variation between neighboring sites within the same object. Λ is used to control the importance of smooth term versus data term.

4. EXPERIMENTAL RESULTS

In this section the results of the conducted experiments with color, motion and texture features for foreground video segmentation is analysed and compared with the work reported by [1]. In [1] the user drew a stroke on the input video frame for a faster discrimination of foreground and background. However the final segmented foreground object contained some portions of the background too. The following figures 3 and 4 shows the input and output of the method put forward by [1].

In our proposed method the CNN automatically segments the foreground image by using the cues viz color, motion and texture. It automatically computes the pixel motions from Ft-1 to ft and ft to ft+1. The randomized texton algorithm helps in finding the correct texture of the foreground. Hence there is no need of an user interaction with the input video frames for separating the foreground and background regions. Figures 5 and 6 shows the frame used in our approach and the corresponding extracted foreground image.

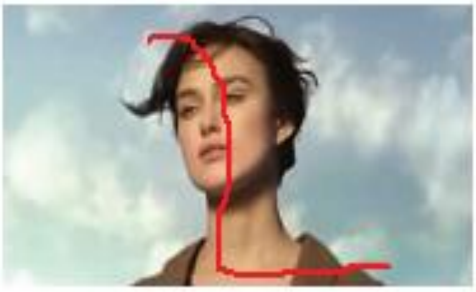

Figure 3.User-stroked frame

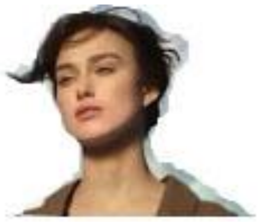

Figure 4.Segmented foreground image with some portions of the background

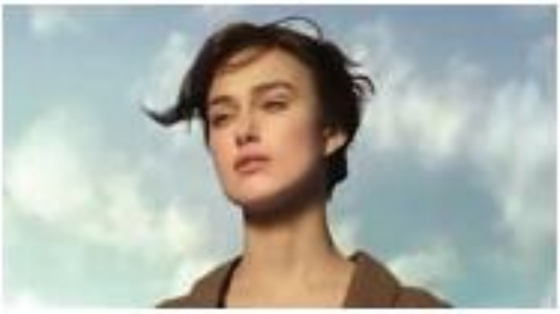

Figure 5.Input video frame 1

The existing work also possesses various difficulties in handling novel foreground colours. For each incoming frames the colour of the foreground may vary and new objects may also occupy parts of frames. Hence when a new unseen pixel is come across in the incoming frame then that pixel is labelled as unknown as shown in figure A1. When final binary segmentation is performed by global optimization, these pixels are labelled as background due to the aforementioned bias. However the algorithm corrects these mistakes only after 10 frames. Hence the rate of misclassification is much higher. Figure A1 shows the misclassification by CSVMs.

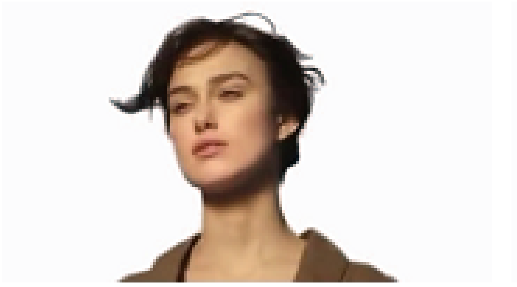

Figure 6.Extracted foreground image from our method

Initially there is no hand motion in the frame. But suddenly when a hand appears in the frame CSVMs misclassify the number of pixels as outliers or as apart which do not belong to the foreground image. But as shown in the consecutive images the misclassifications gradually reduces and it becomes nil in the final image. The final image is the one which is got after 10 iterations. Hence it requires 10 iterations for the existing algorithm to correct the misclassifications.

However in our method which uses CNN the novel foreground colour as well as the unseen pixels are correctly classified by the estimated pixel motion and the randomized texton searching algorithm. Figure A2 shows the output produced by our system for the same input frame used by [11].

Similar to the CSVMs misclassification occurs in the initial frame. However the number of misclassifications is much smaller when compared to CSVMs.

The graph shown in figure A3 depicts the comparison between the error percentages of the existing as well as proposed methods for increasing number of frames.

The higher median error rates of existing and proposed systems are 2.49% and 1.61% respectively for the same sequences. Hence it is clear that the error rate of the proposed method is comparatively lesser than the existing system.

5. CONCLUSION

This paper presented a CNN, texture and graph cut frame work which is able to effectively and efficiently deal with foreground segmentation from live videos. The proposed method is completely robust to relative foreground-background motion of any magnitude and prevents background pixels near the boundary from being misclassified as foreground. For the texture extraction here we introduce new randomized texton searching method and its gives good result in the foreground segmentation. It can also handle difficult scenarios such as camera motion, fuzzy objects, dynamic background, topology changes etc. In future, we can integrate gradient, motion of pixel, colour adjacency, spatial-temporal coherency, etc. to the parameters used viz pixel motion, texture and colour for further improvement of the proposed method.

REFERENCES

[1] C.Wren, A.Azarhayejani, T.Darrell and A.P. Pentland, Pfinder:Real Time Tracking of the Human Body, IEEE Transactions on Pattern Analysis and Machine, Vol. 19, No. 7, 1997, pp. 780-785, http://dx.doi.org/10.1109/34.598236.

[2] C. Stauffer and E. Grimson, Adaptive Background Mixture Models for Real time Tracking, The Artificial Intelligence Laboratory, 1999.

[3] S.Y.Chien, W.K.Chan, Y.H.Tseng and H.Y.Chen, Video Object Segmentation and Tracking Framework with improved Threshold Decision and Diffusion Distance, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 23, No. 6, 2013, pp. 921-934, http://dx.doi.org/10.1109/TCSVT.2013.2242595.

[4] L.Maddalena and A.Petrosino, A Self-Organizing approach to Background Subraction for Visual Surveillance Applications, IEEE Transactions on Image Processing, Vol. 17, No. 7, 2008, pp. 1168-1177, http://dx.doi.org/10.1109/TIP.2008.924285.

[5] K.Kim, T.Chalidabhongse, D.Harwood and L. Davis, Background Modelling and Subtraction by Codebook Construction, Proc.International Conference on Image Processing, Naples, 2004, pp. 3061-3064, http://dx.doi.org/10.1109/ICIP.2004.1421759.

[6] H.Wang and D.Suter, A Consensus based method for Tracking:Modelling Background Scenario and Foreground Appearance, Pattern Recognition, Vol. 40, No. 3, 2007, pp. 1091-1105, http://dx.doi.org/10.1016/j.patcog.2006.05.024

[7] O.Barnich and M.Van Droogenbroeck, Vibe:A Universal Background Subtraction Algorithm for Video Sequences, IEEE Transactions on Image Processing, Vol. 20, No. 6, 2011, pp. 1709-1724, http://dx.doi.org/10.1109/TIP.2010.2101613 .

[8] M.Hofmann, P.Tiefenbacher and G.Rigoll, Background Segmentation with Feedback: The Pixel-Based Adaptive Segmenter, in: Proc. IEEE Computing Society Conference on Computer Vision and Pattern Recognition, Providence, RI, 2012, pp. 38- 43, http://dx.doi.org/10.1109/CVPR.2012.6247657

[9] A.Elgammal, R.Duraiswami, D.Harwood and L.Davis, Background and Foreground Modelling using Nonparametric Kernel Density Estimation for Visual Surveillance Proc. IEEE, USA, 2002, pp. 1151-1163, http://dx.doi.org/10.1109/JPROC.2002.801448

[10] D.Russell and S.Gong, A Highly Efficient Block-Based Dynamic Background Model, Proc. IEEE Conference on Advanced Video and Signal Based Surveillance, USA, 2005, pp. 417-422, http://dx.doi.org/10.1109/TIP.2008.924285

[11] Jess Mathew, Vertical Edge Detection for Car License Plate Recognition, DJ Journal of Advances in Electronics and Communication Engineering, Vol. 1, No. 1, 2015, pp. 8-15, http://dx.doi.org/10.18831/djece.org/2015011002 .

[12] Minglun Gong, Yiming Qian and Li Cheng, Integrated Foreground

Segmentation and Boundary Matting for Live Videos, IEEE Transactions on Image Processing, Vol. 24, No. 4, 2015, pp. 1356-1370, http://dx.doi.org/10.1109/TIP.2015.2401516 .

[13] Wei Ma, Yu Zhang, Luwei Yang and Lijuan Duan, Graph-Cut based Interactive Image Segmentation with randomized Texton Searching, Computer Animation and Virtual Worlds, 2015, http://dx.doi.org/10.1002/cav.1671

[14] J.Rajeesh and E.Arun, Region Growing and Level Set Compound for Hippocampus Segmentation, DJ Journal of Advances in Electronics and Communication Engineering, Vol. 1, No. 1, 2015, pp. 1-7, http://dx.doi.org/10.18831/djece.org/2015011001

APPENDIX A

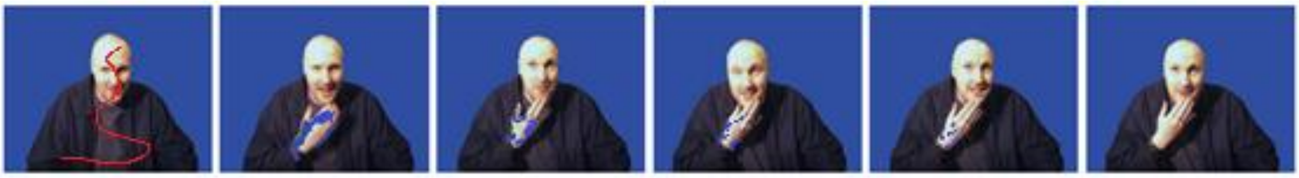

Figure A1.Existing system failure

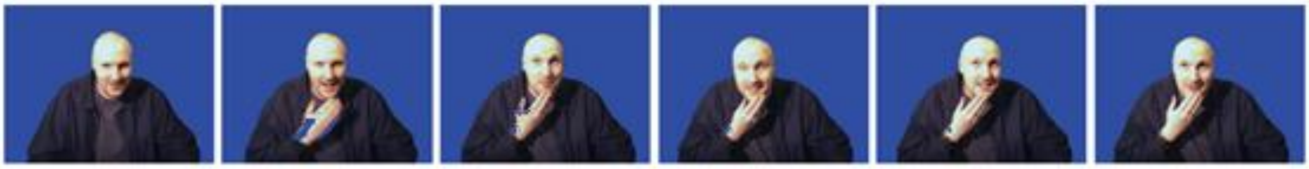

Figure A2.Less number of pixels is misclassified by CNN.

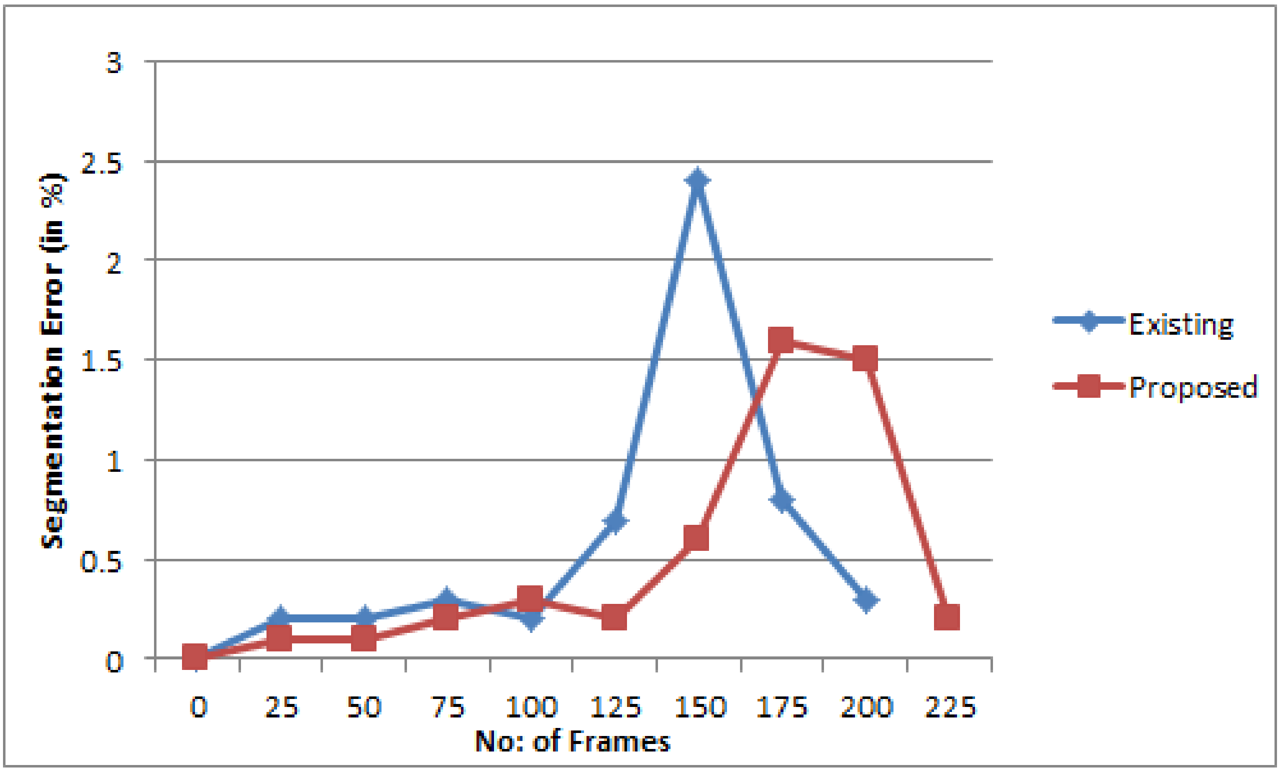

Figure A3.Segmentation error percentage of existing and proposed method