

Strategy for Foreground Movement Identification Adaptive to Background Variations

K. Anuradha, N. R. Raajan

School of Electrical and Electronics Engineering, SASTRA Deemed University, India

Article Info

Article history:

Received Jan 30, 2018

Revised Jul 18, 2018

Accepted Aug 9, 2018

Keyword:

Adaptive threshold
Background subtraction
Frame difference
Motion detection
Tracking

ABSTRACT

Video processing has gained a lot of significance because of its applications in various areas of research. This includes monitoring movements in public places for surveillance. Video sequences from various standard datasets such as I2R, CAVIAR and UCSD are often referred for video processing applications and research. Identification of actors as well as the movements in video sequences should be accomplished with the static and dynamic background. The significance of research in video processing lies in identifying the foreground movement of actors and objects in video sequences. Foreground identification can be done with a static or dynamic background. This type of identification becomes complex while detecting the movements in video sequences with a dynamic background. For identification of foreground movement in video sequences with dynamic background, two algorithms are proposed in this article. The algorithms are termed as Frame Difference between Neighboring Frames using Hue, Saturation and Value (FDFN-HSV) and Frame Difference between Neighboring Frames using Greyscale (FDFN-G). With regard to F-measure, recall and precision, the proposed algorithms are evaluated with state-of-art techniques. Results of evaluation show that, the proposed algorithms have shown enhanced performance.

Copyright © 2018 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

K. Anuradha,
School of Electrical and Electronics Engineering,
SASTRA Deemed University,
Tirumalaisamudram, Thanjavur, India.
Email: kanukalyan79@gmail.com

1. INTRODUCTION

In video processing, foreground identification can be accomplished with a static or dynamic background. This process is difficult, when identifying the movements in video sequences with a dynamic background. Numerous algorithms have performed the foreground identification in video sequences. It is found that, these algorithms have done the foreground identification with less importance to changes in background and illumination. This work proposes two algorithms FDFN-HSV and FDFN-G. The proposed algorithms have identified the foreground movements with significance to variations in illumination and background. The proposed algorithms have addressed the issue, by computing the adaptive threshold of the changes among frames. This section presents the literature of the relevant work studied.

A robust technique for target tracking was presented in [1]. This technique mines the targets from a video. Then, the targets were categorized pertinent to properties of images. This method has tracked objects with variations in appearance. A background model for surveillance with the Pantilt-Zoom (PTZ) camera was proposed in [2]. This model has presented a texture descriptor for encoding spatio-temporal data. This model was linked with a set of techniques for tracking objects. In [3], a textual descriptor referred as Center symmetric scale invariant local ternary patterns (CS-SILTP) was contributed for building a background model. This model was applied for identifying foreground objects in each of the captured frames. To identify

the objects in motion, a non-panoramic algorithm relying on spatio-temporal background was presented in [4]. This algorithm has solved the problems with panoramic methods such as slow initialization, background adaptation and error accumulation. Relying on Bayes decision framework, a technique to excerpt foreground objects from a video was contributed in [5]. In this work, a background with moving and static objects was modeled. Then, the objects in the foreground were identified by combining the outcome of static points and points in movement.

By applying a mixture model, the work proposed in [6] identifies shadows in motion. This technique has reduced overhead by detecting shadows only on foreground pixels. By sampling values, the codebook background subtraction method presented in [7] captures structural background movements for a long time. This algorithm has tracked moving objects with varying background and changes in illumination. To deal with complicated dynamic background, a method was proposed in [8]. This method has proposed a scale-invariant pattern operator and a technique for estimation of kernel density in patterns. By combining these, the method has handled dynamic background. An algorithm for detecting objects in motion was offered in [9]. This method stores the values of pixels in the past. Then, it relates the present and past values of the pixel. Finally, the method identifies the particular background to which the pixel belongs. By combining the color information and SILTP, a block-wise background model was contributed in [10]. This model has dealt with multimodal and dynamic background. For efficient identification of objects in motion, the color information and SILTP were combined.

An efficient method based on background subtraction was presented in [11]. By reducing the data dimensionality of image frame and applying the sparse representation, the method has extracted foreground objects. A new integrated method was proposed in [12]. By combining the region growing with thresholding, the method finds the region of interest (ROI) in an image. An algorithm termed as DTGLMM-K was offered in [13]. The algorithm has focused on image segmentation in an enhanced manner. This algorithm was appropriate for various types of data and applications. The method developed in [14] presents a framework for detection and recognition of human actions. The method segments the objects in motion, excerpts a set of features and chooses the features. For recognition of actions, the chosen features were classified with a multiclass SVM. A method for identifying moving objects was proposed in [15]. The method assesses the bi-directional optical flow between frames. Then it is improved and normalized. Finally, the method identifies the moving objects by verifying the threshold of the optical flow of each block and the optical flow of the object under consideration. An algorithm termed as 'Good Features to Track' was presented in [16]. First, the algorithm extracts feature from the frames. Then, the features of moving objects were excerpted from the subsequent frame. Based on the motion information and location, the objects in motion for each frame were identified. This work has also formed clusters of objects in motion.

Depending on the fixed spatial association among the current and randomly chosen pixels, a novel framework was presented in [17]. The method forms a sample set of spatial information for each pixel. Then, the Spatial Sample Difference Consensus (SSDC) was computed for identification of foreground objects. An approach for detection of motion and segmentation was developed in [18]. The approach controls the uncertainties in the movement of the camera and computational disparities. By combining the color, motion probability and depth cues, the approach has segmented the moving objects. With importance to statistical learning, an efficient method for image segmentation was presented in [19]. The method applies the Rayleigh distribution to compute the probability density of background pixel. This method was applied to images of various colors. An algorithm based on the scale invariant feature transform (SIFT) was contributed in [20]. The method was applied for identifying gradual transitions and sudden variations without a need for training of the video. A method for detection of moving objects in GPU was proposed in [21]. This method has enhanced the quality of outcome in scenarios where the background and the objects in motion appear to be same. This method has automatically chosen the regions of interest (ROI).

2. RESEARCH METHOD

Two algorithms FDNF-HSV and FDNF-G are proposed for foreground movement identification. The FDNF-HSV algorithm identifies the foreground movement using HSV color model. Section 2.1.1 presents the algorithm for FDNF-HSV. Section 2.2.1 presents the algorithm for FDNF-G. The algorithms common for FDNF-HSV and FDNF-G (i.e.) calculation of threshold and drawing bounding box are presented in sections 2.1.2 and 2.1.3 respectively.

2.1. Motion detection using HSV channel

The proposed method referred as FDNF-HSV (Frame Difference between Neighboring Frames using HSV) focuses on the foreground detection. The proposed algorithm works as follows: Using HSV (Hue, Saturation, Value) color model, a pixel-based difference among the frames are computed. For each frame taken into account, the three components of the frame (i.e.) HSV are obtained. Based on the HSV, the

difference among the successive frames (i.e.) del_h , del_s and del_v are computed. Subsequently, three binary images are obtained by applying the proposed adaptive threshold algorithm presented in 2.1.2. Using equation (1), the resultant image is obtained by calculating the maximum value from each pixel of binary images.

$$result_f = \max(th_im, th_im_1, th_im_2) \quad (1)$$

After filtering the small connected regions in the $result_f$, the proposed algorithm detects the foreground movement by applying the bounding box method.

2.1.1. Algorithm for Motion Detection using HSV

Step 1 : $In_b = frameRGBtoHSV(b)$ // changes the true color image RGB to the HSV of background image b

Step 2 : for $f = 2$ to $NF-1$ // NF denotes the number of frames

Step i : $In_f = frameRGBtoHSV(f)$ (i.e.) $In_f = \{In(i, j) | 1 \leq i \leq row, 1 \leq j \leq col\}$
// changes the true color image RGB to HSV color channels

Step ii : Similarly $In_{f+1} = frameRGBtoHSV(f+1)$

Step iii : Find the absolute difference between H channel In_{f+1} and In_f :

$$del_h = |In_{f+1}(:, :, 1) - In_f(:, :, 1)|$$

Step iv : Find the absolute difference between S channel In_{f+1} and In_f :

$$del_s = |In_{f+1}(:, :, 2) - In_f(:, :, 2)|$$

Step v : Find the absolute difference between V channel In_{f+1} and In_f :

$$del_v = |In_{f+1}(:, :, 3) - In_f(:, :, 3)|$$

Step vi : Call $th_im = \text{Threshold}(del_3, del_3)$

Step vi : Call $th_im1 = \text{Threshold}(del_1, del_1)$

Step vii : Call $th_im2 = \text{Threshold}(del_2, del_2)$

Step viii : Find the result by finding pixel-wise the maximal value

$$result_f = \max(th_im, th_im_1, th_im_2)$$

Step ix : Convert image $result_f$ to Boolean image by applying

$$bool(i, j) = \begin{cases} T : result_f(i, j) = 1 \\ F : result_f(i, j) = 0 \end{cases}$$

$$\forall 1 \leq i \leq row, 1 \leq j \leq col$$

Step x : Draw bounding box to detect the foreground movement.

2.1.2. Algorithm for Threshold calculation: Threshold (im_1, im_2)

Step 1 : Calculate threshold using $\Theta = \frac{1}{row + col} \left(\sum_{col} \max(im_1) + \sum_{row} \max(im_1^T) \right)$

Step 2 : If $\Theta \neq 0$ then binary image obtained is

$$timage_f = \{pix(i, j) | 1 \leq i \leq row, 1 \leq j \leq col\}, \text{ where}$$

$$pix(i, j) = \begin{cases} 0, \forall im_1(i, j) \in im_1 : im_1(i, j) \leq \Theta \\ 1, otherwise \end{cases}$$

Step 3 : If $\Theta = 0$ then binary image obtained is

$$timage_f = \{pix(i, j) | 1 \leq i \leq row, 1 \leq j \leq col\}, \text{ where}$$

$$pix(i, j) = \{im_2(i, j) | 1 \leq i \leq row, 1 \leq j \leq col\}$$

2.1.3. Algorithm for Drawing Bounding Box

- Step 1 : Find the connected regions in the given Boolean image.
 Step 2 : Filters out smaller connected regions
 Step 3 : Identify the four corners for each region, i.e. blobs
 Step 4 : Mark blobs found in the frame with rectangles.

2.2. Motion Detection using Grey scale

The proposed algorithm termed as FDNF-G (Frame Difference between Neighboring Frames using Grey scale) combines the background subtraction with the pixel-based variation between neighboring frames. The algorithm starts by finding the grey scale for the frames in the video sequence under consideration. Three different images del_1 (absolute difference between the current frame and its previous frame), del_2 (absolute difference between the current frame and its next frame) and del_3 (absolute difference between the current frame and the background frame) are obtained. By applying the proposed adaptive threshold algorithm 2.1.2, the binary images for del_1 , del_2 and del_3 are obtained. After filtering the noise, the algorithm identifies the foreground motion. Algorithm for motion detection using greyscale:

Step 1 : $In_b = frameRGBtoGrey(b)$ // changes the true color image RGB to the grayscale intensity of background image b

Step 2 : Find binary image for In_b as bin_b

Step 2 : for $f = 2$ to NF-1 // NF denotes the number of frames

Step i : $In_f = frameRGBtoGrey(f)$ (i.e.) $In_f = \{In(i, j) | 1 \leq i \leq row, 1 \leq j \leq col\}$ // changes the true color image RGB to the grayscale intensity of frames.

Step ii : Similarly $In_{f-1} = frameRGBtoGrey(f-1)$ $In_{f+1} = frameRGBtoGrey(f+1)$

Step iii : Find the absolute difference of In_{f-1} and In_f : $del_1 = |In_{f-1} - In_f|$

Step iv : Find the absolute difference of In_{f+1} and In_f : $del_2 = |In_{f+1} - In_f|$

Step v : Find the absolute difference of In_b and In_f : $del_3 = |In_b - In_f|$

Step vi : Call $th_im = Threshold(del_3, bin_b)$

Step vii : Call $th_im1 = Threshold(del_1, th_im)$

Step viii : Call $th_im2 = Threshold(del_2, th_im)$

Step ix : Find the result by finding pixel-wise logical or operation

$$result_f = th_im | th_im1 | th_im2$$

Step x : Convert image $result_f$ to Boolean image by applying

$$bool(i, j) = \begin{cases} T : result_f(i, j) = 1 \\ F : result_f(i, j) = 0 \end{cases}$$

$$\forall 1 \leq i \leq row, 1 \leq j \leq col$$

Step xi : Draw bounding box to detect actor

3. RESULTS AND DISCUSSION

The proposed algorithms FDNF-HSV and FDNF-G have been assessed with the datasets viz. UCSD, I2R and CAVIAR. The outcome of the assessments is shown in Figure 1. The proposed algorithm was assessed with various metrics viz. recall, precision, F-measure and percentage of wrong classification (pwc). The following parameters were considered for computing these metrics.

- True positive (TP)-pixels in a foreground object categorized as pixels in the foreground.
- True negative (TN)-pixels in a background object categorized as pixels in the background.

- c. False positive (FP)-pixels in a background categorized as pixels in the foreground.
- d. False negative (FN)-pixels in a foreground categorized as pixels in the background.



Figure 1. Sample output for the UCSD dataset, I2R dataset and CAVIAR dataset

The calculation of pwc, precision, recall and F-measure (similarity measure) are given here.

$$pwc = \frac{FN + FP}{TP + FN + FP + TN} \times 100\% \quad pre = \frac{TP}{TP + FP} \quad recall = \frac{TP}{TP + FN}$$

$$F_{measure} = \frac{2 \times recall \times pre}{recall + pre}$$

With reference to I2R, CAVIAR and UCSD datasets, the proposed algorithms FDNF-HSV and FDNF-G detect activities with complex differences in the background. This encompasses illumination variations and acquiring dynamic activities. The I2R dataset comprises video sequences for identification of foreground objects with intricate background. Pertinent to F-measure metric for the I2R dataset, an assessment of the proposed approach (FDNF-HSV) with other techniques is presented in Table 1. Results exhibit that, the proposed approach has accomplished maximum F-measure than the other techniques. Figure 2 provides a graphical representation of the results attained for depicting the metrics viz., precision, recall, F-measure for I2R dataset. Figure 3 presents a graphical representation of the outcome obtained for the metrics viz., precision, recall, F-measure for CAVIAR dataset.

Table 1. Assessment of F-measure with various techniques and the proposed method (FDNF-HSV) for the I2R dataset

	Bootstrap	Escalator	Fountains	Lobby	Water surface
FrmDiff[1]	35.36	21.72	25.43	16.33	24.26
MoG[6]	56.29	41.17	76.91	47.92	79.7
ACMMM03[5]	60.43	32.6	56.51	30.31	63.66
Code Book [7]	63.66	49.82	61.36	25.51	73.09
SILTP[8]	73.32	65.88	86.23	78.57	84.36
ViBe[9]	78.26	64.72	60.96	26.55	86.82
CS-SILTP[3]	76.35	70.72	87.46	80.23	87.38
BITC[10]	64.86	63.37	95.24	76.67	93.02
DFB -A[4]	71.86	66.37	77.43	13.24	93.81
Non-Parametric[21]	64.10	-	70.49	-	90.11
Proposed Approach (FDNF-HSV)	86.48	78.04	86.69	97.43	98.98

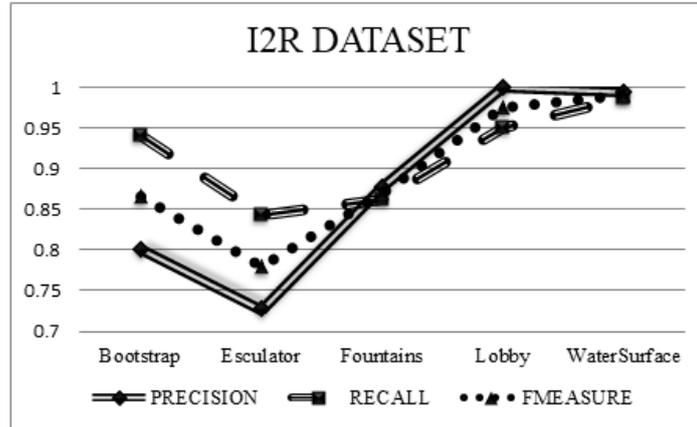


Figure 2. Graph depicting metrics viz., precision, recall, F-measure for I2R dataset

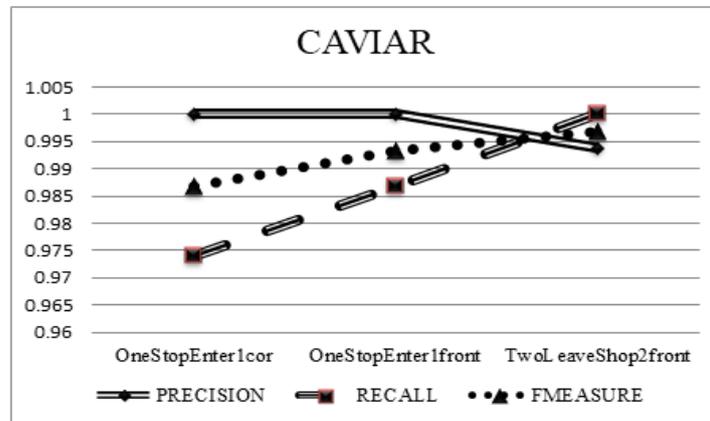


Figure 3. Graph representing metrics viz., precision, recall, F-measure for CAVIAR dataset

The proposed method (FDFN-G) aims at identifying foreground movement in video sequences available in UCSD background subtraction dataset. The results of validation are presented in Table 2. Results illustrate that, the proposed method has attained an improved F-measure when compared with other methods taken into account.

Table 2. Comparison of the proposed technique (FDFN-G) with various methods on UCSD background subtraction dataset with regard to average F-measure

Sequences	FrmDiff [1]	MoG [6]	ACMMM03 [5]	Code Book [7]	SILTP [8]	ViBe [9]	CS-SILTP [3]	Proposed Work (FDFN-G)
Birds	19.63	28.35	28.82	21.21	30.11	31.92	29.76	44.44
Bottle	19.31	51.12	23.44	23.09	55.23	57.5	63.41	100
Freeway	26.81	53.06	31.88	38.48	51.94	54.62	55.54	86.41
Ocean	10.07	28.95	19.03	22	57.87	26.33	60.63	94.11
Pedestrians	24.28	79.63	28.89	64.3	81.09	80.1	85.7	94.73
Rain	41.88	74.81	81.28	43.28	85.24	93.7	89.74	94.34
AvgF-measure	23.66	52.65	35.56	35.39	60.25	57.36	64.13	85.67

4. CONCLUSION

Research on various methods and algorithms in video processing is commendable. Relevant to the identification of foreground in video sequences, lot of algorithms are available in the literature. Most of these algorithms have done the foreground identification with less attention to variations in illumination and background. But, the proposed algorithms FDFN-HSV and FDFN-G have identified the foreground

movement with importance to changes in illumination and background. The proposed work has performed foreground movement identification by calculating the adaptive threshold of the variations between frames. Results of the proposed algorithms are better than the other standard algorithms in terms of similarity metrics.

REFERENCES

- [1] A. J. Lipton, *et al.*, "Moving target classification and tracking from real-time video," *4th IEEE Workshop on Applications of Computer Vision*, vol/issue: 98(2), pp. 8–14, 1998.
- [2] N. Liu, *et al.*, "Hierarchical ensemble of background models for PTZ-based video surveillance," *IEEE transactions on cybernetics*, vol/issue: 45(1), pp. 89–102, 2015.
- [3] H. Wu, *et al.*, "Real-time background subtraction-based video surveillance of people by integrating local texture patterns," *Signal, Image and Video Processing*, vol/issue: 8(4), pp. 665–676, 2014.
- [4] S. W. Kim, *et al.*, "Detection of moving objects with a moving camera using non-panoramic background model," *Machine Vision and Applications*, vol/issue: 24(5), pp. 1015–1028, 2013.
- [5] L. Li, *et al.*, "Foreground object detection from videos containing complex background," *Proceedings of the eleventh ACM international conference on Multimedia*, pp. 2–10, 2003.
- [6] P. Kaewtrakulpong and R. Bowden, "An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection," *Advanced Video Based Surveillance Systems*, pp. 1–5, 2001.
- [7] K. Kim, *et al.*, "Real-time foreground- background segmentation using codebook model," *Real-Time Imaging*, vol/issue: 11(3), pp. 172–185, 2005.
- [8] S. Liao, *et al.*, "Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes," *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 1301–1306, 2010.
- [9] O. Barnich and M. Van Droogenbroeck, "ViBe: A universal background subtraction algorithm for video sequences," *IEEE Transactions on Image Processing*, vol/issue: 20(6), pp. 1709–1724, 2011.
- [10] H. Han, *et al.*, "Moving object detection revisited: Speed and robustness," *IEEE Transactions on Circuits and Systems for Video Technology*, vol/issue: 25(6), pp. 910–921, 2015.
- [11] Y. Wang, *et al.*, "Compressive background modeling for foreground extraction," *Journal of Electrical and Computer Engineering*, vol. 2015, pp. 1–8, 2015.
- [12] S. Hore, *et al.*, "An integrated interactive technique for image segmentation using stack based seeded region growing and thresholding," *International Journal of Electrical and Computer Engineering*, vol/issue: 6(6), pp. 2773–2780, 2016.
- [13] T. Jyothirmayi, *et al.*, "Image segmentation based on doubly truncated generalized laplace mixture model and K means clustering," *International Journal of Electrical and Computer Engineering*, vol/issue: 6(5), pp. 2188–2196, 2016.
- [14] M. Sharif, *et al.*, "A framework of human detection and action recognition based on uniform segmentation and combination of Euclidean distance and joint entropy-based features selection," *Eurasip Journal on Image and Video Processing*, vol/issue: 2017(1), 2017.
- [15] S. S. Sengar and S. Mukhopadhyay, "Motion detection using block based bi-directional optical flow method," *Journal of Visual Communication and Image Representation*, vol. 49, pp. 89–103, 2017.
- [16] A. Keivani, *et al.*, "Motion-based moving object detection and tracking using automatic K-means," *AFRICON, 2017 IEEE*, pp. 32–37, 2017.
- [17] C. Zhang, *et al.*, "Moving object detection algorithm based on pixel spatial sample difference consensus," *Multimedia Tools and Applications*, vol/issue: 76(21), pp. 22077–22093, 2017.
- [18] D. Zhou, *et al.*, "Moving object detection and segmentation in urban environments from a moving platform," *Image and Vision Computing*, vol. 68, pp. 76–87, 2017.
- [19] M. Moussa, *et al.*, "Comparative study of statistical background modeling and subtraction," *Indonesian Journal of Electrical Engineering and Computer Science*, vol/issue: 8(2), pp. 287–295, 2017.
- [20] Y. Tabii, *et al.*, "Video Shot Boundary Detection using the Scale Invariant Feature Transform and RGB Color Channels," *International Journal of Electrical and Computer Engineering (IJECE)*, vol/issue: 7(5), pp. 2565–2673, 2017.
- [21] D. Berjón, *et al.*, "Real-time nonparametric background subtraction with tracking-based foreground update," *Pattern Recognition*, vol. 74, pp. 156–170, 2018.