A Novel Method to Improve Basic Background Subtraction Methods for Object Detection in Video Surveillance System

Surender Singh  
Research Scholar, School of Computer Science and Engineering, University of Petroleum and Energy Studies, Dehradun, India.  
Orcid Id: 0000-0002-7996-8588

Ajay Prasad  
Associate Professor, School of Computer Science and Engineering, University of Petroleum and Energy Studies, Dehradun, India.

Kingshuk Srivastava  
Assistant Professor, School of Computer Science and Engineering, University of Petroleum and Energy Studies, Dehradun, India.

Suman Bhattacharya  
Project Head, IPR Management Services, Tata Consultancy Services, Bhubaneswar, India.

Abstract
This paper proposes a novel method for the improvement of basic Background Subtraction (BGS) methods to detect moving objects in video surveillance streams. The method is based on Local Neighborhood Differencing (LND) in which instead of finding a simple pixel to pixel difference between current frame and background model, the average of the pixel neighborhoods from the current frame and background model are subtracted to entitle the pixel a background or foreground in the current frame in order to find moving objects in video. The proposed method has been tested on two basic methods; Adaptive Mean and Adaptive Median methods of object detection using various complex real time benchmarked scenarios. It is also compared with classical statistical thresholding method. The results have been measured in precision and recall metrics to register improvement. The obtained results have confirmed the utility of the method by increasing the robustness of the object detection techniques in video surveillance for real time video analytic.

Keywords: Object Detection, Motion Detection, Background Subtraction, Automated Video Surveillance, Adaptive Mean, Adaptive Median, P-R Curves.

INTRODUCTION
Real-time video surveillance systems detect objects or situations in video flow that represent a security threat and trigger an alarm accordingly. These systems can be classified into operator controlled, automated video surveillance and intelligent video surveillance systems [19]. In operator controlled surveillance system, the video stream is analyzed manually; a person observes the video to determine if there is any activity that requires an action. In the second approach, the automated video surveillance system uses motion detection techniques to determine response. An intelligent video surveillance system is that which extract the relevant information from generic motion accurately and issue actions. Existing video surveillance systems take care about video capture, store and transmission of video to remote places but devoid of efficient object detection and analysis leaving these functions exclusively to human operators for manual analysis [6]. Therefore, there is an urgent need of a surveillance system which is fast, efficient and accurate.

There are several categories of object detection methods out of which the background subtraction is most popular and traditionally used category [18]. In this category, there are robust methods such as Kernel Density Estimation and Histogram Detection which provide reliable detection but these are also slow and less useful for real time analytics. This category also includes some basic methods like Frame Differencing, Adaptive Mean, Adaptive Median methods which are fast but do not provide good object detection results. Our research is targeted to improve these basic BGS methods for object detection.

This paper is organized as follows: Section 1 identifies the background on the need of modification in object detection paradigm to make it more robust and useful for real time scenarios. Section 2 presents the related works which made efforts to improve the basic methods. Section 3 describes the proposed methodology. Section 4 tabulates and compares the results obtained with the proposed methods and other past improvements in basic methods. Section 5 discusses the results which is followed by conclusion and future scope in section 6.

RELATED WORKS
Most of the improvements made in the past in basic background subtraction methods revolves around proposing a threshold which is effective and adaptive in different situations or scenarios [1]. Many statistical measures such as mean, median, deviation, outliers and variance from mean are used to define a
new adaptive and automatic threshold based on two-frame, three-frame and frame-background differencing [5].

Sezgin and Sankur categorized thresholding methods into various groups based on different thresholding criterion [15] such as histogram component based methods, clustering-based methods, entropy-based methods, attribute based, spatial and local method. Histogram based methods uses various components of histograms; peaks, valleys and curvature to derive adaptive threshold. Nain et al proposed histogram based thresholding method by finding prominent peaks to represent the distinct regions in the image [12].

Cluster-based methods model threshold as Gaussians function or multimodal distribution. Hua et al proposed an adaptive threshold for non-parametric Kernel Density Estimation to address the bimodal intensity distribution video sequences for object detection [9]. Lai and Rosin demonstrated that circular Otsu method based adaptive thresholding scheme provides better result than the linear Otsu thresholding [10]. Entropy-based methods use the entropy of the foreground and background of the image for determining threshold for object detection. Subudhi et al proposed an entropy based adaptive threshold for spatial segmentation and temporal segmentation based object detection algorithm [17].

Object attribute-based methods measure similarity between the gray-level and the binary images for finding optimum threshold. Samanta and Sanyal defined adaptive threshold by using mean and variance of neighborhood of each pixel. Spatial methods use correlation or probability distribution for adaptive threshold and local methods may adapt threshold value based on local characteristics of each pixel in the image [14]. The work in [4] proposed an adaptive threshold using ROC curve by averaging block thresholds.

Zidek and Hosovsky [20] further classified thresholding into static(fixed) and dynamic(adaptive) thresholding. They also reported that hybrid thresholding based on multispectral and Otsu method provides better detection result. [Rosin] Rosin et al classified global thresholding into Euler-number based on a fix number for every stable block of image, Poisson-noise modelling based on Poisson model for every pixel and entropy method based on the change in intensity.

Firdousi, and Parveen compared Niblack’s, Yanowitz and Bruckstein’s, Bernsen’s Techniques to conclude that the local gray range method performs better than local variance methods [7]. Boufares et al proposed wavelet transforms induced adaptive threshold for object detection in BGS methods [2]. Isaac developed an adaptive threshold method by using second derivative of cumulative sum of difference frame [3]. Authors in [16] also proposed a novel local adaptive thresholding scheme based on local average of intensities to evolve edge map of images which provided better result as compared to global/fixed threshold.

As discussed above, there are many methods proposed in the past which have exploited local neighborhood of a pixel to determine adaptive threshold only but to the best of our knowledge, not a single method or technique has been proposed to compute difference based on local neighborhood. Our hypothesis is that Local Neighborhood Difference (LND) technique can better address noise in object detection process without taking much additional time.

Existing BGS Methods

Most used and traditional BGS method is frame differencing (FD) in which moving objects in a video is obtained by thresholding the difference of time adjacent frames but unfavorable frame rate may fail to detect moving objects.

\[
Diff(x, y) = F_t(x, y) - F_{t-1}(x, y) 
\]

\[
M_t(x, y) = \begin{cases} 
1 & \text{if } abs(Diff(x, y)) \geq Th \\
0 & \text{otherwise} 
\end{cases} 
\]

Where \(F_t(x, y)\) is current frame intensity \((x, y)\) pixel. \(M_t(x, y)\) is object detection at \((x, y)\) thresholded with a fixed value \(Th\). This value can be fixed based on empirical experience or can be model as a percent of average intensity of image. An optimal threshold is very important for better result in these methods.

The FD method can be improved by modeling a background using some initial frames which may detect moving object accurately by finding difference between current frame and background and then thresholding using (2).

\[
Diff(x, y) = F_t(x, y) - B(x, y) 
\]

The background \(B\) has to be updated continuously either with running mean (Adaptive Mean – AM) or running median (Adaptive Median- AMD) methods to take care of dynamicity in video scenes.

**AM Method:**

\[
B_t(x, y) = \epsilon \times F_t(x, y) + (1 - \epsilon) \times B_{t-1}(x, y) 
\]

where \(\epsilon\) is frame refreshing rate and \(t\) is temporal dimension.

**AMD Method:**

\[
B_t(x, y) = B_{t-1}(x, y) + 1 \text{ if } Diff(x, y) > 0 \\
B_t(x, y) = B_{t-1}(x, y) - 1 \text{ if } Diff(x, y) < 0 
\]

The Methodology- Local Neighborhood Differencing (LND)

In the above described methods, the difference between two consecutive frames (in case of FD) or between the current frame \(F\) and background \(B\) is computed using the corresponding pixels \((x, y)\) only as shown in (1) and (3). The LND based methodology suggests to consider 9-neighborhood of a pixel while taking difference. Instead of subtracting two corresponding pixels in CF and BG, it requires to subtract average of 9-neighborhood of corresponding pixels to
To make it more clear, let us take an example in which matrix $F$ is current image of 8×8 and $B$ is Background of 8×8 and a duplicate border makes these images of 10×10 size for LND processing as shown in fig 1 and 2 respectively.

$$LND(x, y) = \frac{1}{9} \sum_{i=x-1}^{x+1} \sum_{j=y-1}^{y+1} \text{abs}(F(i,j) - B(i,j))$$

(7)
Simple difference can be simply pixel to pixel difference between two images, but LND suggests that in order to calculate the difference of a particular pixel say 3\textsuperscript{rd} row and 4\textsuperscript{th} column (3, 4) we must find the difference of corresponding 9-neighborhood of the pixel and then averaging it out to determine difference. For example, the simple absolute difference $d$ of pixel (3, 4) is

\[\text{Diff}(3,4) = d(3,4) = \text{abs}(42 - 62) = 20 \] (8)

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>12</td>
<td>18.89</td>
<td>18.00</td>
<td>21.33</td>
<td>22.44</td>
<td>20.22</td>
<td>20.44</td>
<td>21.22</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>21.44</td>
<td>19.56</td>
<td>19.11</td>
<td>22.00</td>
<td>23.00</td>
<td>20.78</td>
<td>20.67</td>
<td>20.78</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>17.56</td>
<td>17.78</td>
<td>18.78</td>
<td>21.89</td>
<td>23.00</td>
<td>21.89</td>
<td>21.89</td>
<td>21.89</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>15.56</td>
<td>18.00</td>
<td>19.78</td>
<td>20.33</td>
<td>19.56</td>
<td>19.33</td>
<td>19.78</td>
<td>20.33</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>12.00</td>
<td>17.11</td>
<td>19.89</td>
<td>18.56</td>
<td>15.33</td>
<td>19.56</td>
<td>16.11</td>
<td>17.44</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>10.00</td>
<td>16.11</td>
<td>18.56</td>
<td>15.78</td>
<td>11.22</td>
<td>10.11</td>
<td>11.00</td>
<td>11.89</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>6.78</td>
<td>11.11</td>
<td>13.00</td>
<td>12.56</td>
<td>10.00</td>
<td>8.44</td>
<td>8.33</td>
<td>7.78</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>4.89</td>
<td>6.78</td>
<td>7.89</td>
<td>9.22</td>
<td>9.11</td>
<td>7.67</td>
<td>6.78</td>
<td>5.00</td>
<td></td>
</tr>
</tbody>
</table>

![Figure 3. LND of current Frame F and current Background B](image)

LND is average difference of 9-neighborhood of (3, 4) pixel which is shown in gray color in figure 1 and 2. According to (7), the average difference of pixel (3,4) is calculated as

\[ LND(3,4) = \frac{d(2,3)+d(2,4)+d(2,5)+d(3,3)+d(3,4)+d(3,5)+d(4,3)+d(4,4)+d(4,5)}{9} \] (9)

i.e.

\[ LND(3,4) = \frac{11 + 16 + 25 + 15 + 20 + 23 + 17 + 22 + 23}{9} = 19.11 \]

**RESULTS AND DISCUSSION**

The proposed method has been tested on a benchmark data set “CDnet2012” [8] using MATLAB programming on an Intel i3 4GB system. The dataset contains six different scenarios of video surveillance system depicting various problems of object detection such as dynamics of scene, occlusion, shadow, ghost, interleaved movements etc. Each scenario includes several video sequence and for experiment purpose, we have chosen six sequences namely Highway (baseline), Badminton (camera jitter), Fountain02 (dynamic background), Sofa (intermittent object motion), Bus station (shadow) and park (thermal imagery) scenes. Some initial frames are used from each sequence to model an initial background image. The results are calculated using recall, precision, F1-measure [11] by comparing groundtruths. Absolute time is measured to compare various algorithms. Twelve thresholds are used to draw P-R curves which represents the quality of detection in the form of area covered by these curves. Higher covered area signifies better result.

Two basic BGS methods; AM and AMD are tested for six sequences for the proposed method to generate total 12 P-R charts. In each chart, four algorithms namely basic BGS with fixed thresholding (Basic\_AM\_FT), basic BGS with adaptive thresholding (Basic\_AM\_AT), LND based BGS with fixed thresholding (LND\_AM\_FT) and LND based BGS with adaptive thresholding (LND\_AM\_AT) are depicted. Figure 4 to figure 9 are for AM methods and figure 10 to figure 15 are for AMD methods.

<table>
<thead>
<tr>
<th>Scenarios</th>
<th>Basic_AM_FT</th>
<th>Basic_AM_AT</th>
<th>LND_AM_FT</th>
<th>LND_AM_AT</th>
<th>% Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>High Way</td>
<td>0.7857</td>
<td>0.7869</td>
<td>0.8225</td>
<td>0.8237</td>
<td>5%</td>
</tr>
<tr>
<td>Badminton</td>
<td>0.5719</td>
<td>0.5718</td>
<td>0.5751</td>
<td>0.5750</td>
<td>1%</td>
</tr>
<tr>
<td>Fountain02</td>
<td>0.7275</td>
<td>0.7274</td>
<td>0.7712</td>
<td>0.7719</td>
<td>6%</td>
</tr>
<tr>
<td>Sofa</td>
<td>0.4688</td>
<td>0.4692</td>
<td>0.4825</td>
<td>0.4824</td>
<td>3%</td>
</tr>
<tr>
<td>Bus Station</td>
<td>0.7176</td>
<td>0.7155</td>
<td>0.7214</td>
<td>0.7183</td>
<td>1%</td>
</tr>
<tr>
<td>Park</td>
<td>0.7052</td>
<td>0.7058</td>
<td>0.7188</td>
<td>0.7211</td>
<td>2%</td>
</tr>
</tbody>
</table>

Table 1. Comparison of Basic AM and LND\_AM method with fixed Threshold (FT) and Adaptive Threshold(AT) using F1 score for the best threshold value
Table 2. Comparison of Basic AMD and LND_AMD method with fixed Threshold (FT) and Adaptive Threshold(AT) using F1 score for the best threshold value

<table>
<thead>
<tr>
<th>Scenarios</th>
<th>Basic_AM_FT</th>
<th>Basic_AM_AT</th>
<th>LND_AM_FT</th>
<th>LND_AM_AT</th>
<th>% Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>High Way</td>
<td>0.8558</td>
<td>0.8565</td>
<td>0.8879</td>
<td>0.8891</td>
<td>4%</td>
</tr>
<tr>
<td>Badminton</td>
<td>0.5677</td>
<td>0.5675</td>
<td>0.5679</td>
<td>0.5677</td>
<td>0%</td>
</tr>
<tr>
<td>Fountain02</td>
<td>0.6960</td>
<td>0.7096</td>
<td>0.7436</td>
<td>0.7522</td>
<td>7%</td>
</tr>
<tr>
<td>Sofa</td>
<td>0.4948</td>
<td>0.4988</td>
<td>0.5201</td>
<td>0.5222</td>
<td>5%</td>
</tr>
<tr>
<td>Bus station</td>
<td>0.7124</td>
<td>0.7130</td>
<td>0.7357</td>
<td>0.7247</td>
<td>3%</td>
</tr>
<tr>
<td>Park</td>
<td>0.7435</td>
<td>0.7421</td>
<td>0.7593</td>
<td>0.7569</td>
<td>2%</td>
</tr>
</tbody>
</table>

Table 1 and 2 display F1 score for all the four type variations of Adaptive Mean and Adaptive Median BGS methods respectively. Last column of each table register the percentage improvement achieved between basic method and LND based method with fixed threshold.

Figure 4: Precision Recall Curve for Adaptive Mean object detection methods for Highway dataset

Figure 5: Precision Recall Curve for Adaptive Mean object detection methods for Badminton dataset

Figure 6: Precision Recall Curve for Adaptive Mean object detection methods for Fountain Dataset data set

Figure 7: Precision Recall Curve for Adaptive Mean object detection methods for Sofa data set

Figure 8: Precision Recall Curve for Adaptive Mean object detection methods for Bus Station data set
Figure 9: Precision Recall Curve for Adaptive Mean object detection methods for Park data set

In most of the scenarios, there is only minor improvement if basic fixed threshold BGS and adaptive threshold basic BGS methods are compared. There is very minor improvement with global adaptive threshold, but when LND method is used in basic BGS method, remarkable improvement ranging from 4% to 7% is achieved in both Mean and Median methods for baseline and dynamic background sequence. For other sequences the improvement is between 1% to 3%. Camera Jitter is the only scenario which doesn’t register any improvement due to definite multimodal background which is difficult to address by single modal methods such as AM and AMD.

Figure 10: Precision Recall Curve for Adaptive Median object detection methods for Highway data set

Figure 11: Precision Recall for Adaptive Median object detection methods for Badminton data set

Figure 12: Precision Recall Curve for Adaptive Median object detection methods for Fountain Dataset data set

Another observation is that there is only a minor performance improvement between fixed threshold and adaptive threshold based BGS detection methods. We have used only a classical global adaptive threshold which is taken as a fraction of average of the current frame in each iteration. Other improved technique of threshold discussed in the related work may provide better results and this need to be investigated further.

Figure 13: Precision Recall Curve for Adaptive Median object detection methods for Sofa data set

Figure 14: Precision Recall Curve for Adaptive Median object detection methods for Bus Station data set
The proposed LND scheme is somewhat less efficient if time constraint is considered in real time videos surveillance system. In these experiments, the LND difference has been calculated by processing frames on pixel level. To make this scheme efficient, we must devise a frame level processing of the LND scheme.

CONCLUSION

This paper proposes a novel scheme of “local neighborhood differencing” for finding difference between the current frame and the current background image in BGS methods. It further validates the proposed methodology empirically on six sequences of a benchmarked dataset. Improvements in the range of 3% to 7% have been registered by the LND scheme as compared to basic methods in single modal background scenarios. It also investigated the effect of global adaptive threshold in foreground detection as compared to fixed threshold. It also identified that the LND scheme needs to be further improved by devising a method which works at frame level processing to be employable in real time video surveillance and analytics.

REFERENCES


[17]. B. N. Subudhi, S. Ghosh, P. K. Nanda, and A. Ghosh,

