International Journal of Mechatronics, Electrical and Computer Technology (IJMEC)
ISSN: 2305-0543 (Online), ISSN: 2411-6173 (Print)
Vol. 10(38), Oct. 2020, PP. 4752-4761

An Enhanced Background Subtraction Algorithm for Smart Surveillance System Using Adaptive Gaussian Mixture Technique

Olayemi M. Olaniyi, Jibril A. Bala, Shefiu Olusegun Ganiyu and Praise Emmanuel. Wisdom

Federal University of Technology Minna, Niger State, Nigeria

**Corresponding Author's E-mail: mikail.olaniyi@futminna.edu.ng*

***Abstract*— Surveillance is an important component of secure infrastructure. People need to ensure that their properties, homes, friends, families, and valuables are protected from intruders. In this article, we developed an enhanced background subtraction algorithm for smart surveillance of homes under changes in illumination. An adaptive Gaussian Mixture technique was applied to improve the conventional background subtraction technique for smart surveillance system by reducing false positives and improving the system performance in motion detection. The developed smart surveillance system has the ability to send notifications containing intruder's image to registered email account and short message system to registered phone number whenever motion is detected. The performance evaluation of developed algorithm showed a sensitivity of 80%, specificity of 92% and an accuracy of 95.56%. This system works in conditions of illumination changes and can be adopted in motion detection for home surveillance..**

Keywords*— Background Subtraction, Motion detection, Receiver Operating Characteristics, Short Message Service, Specificity, Sensitivity, Surveillance, Smart

I. INTRODUCTION

As an essential component of many security associations' priorities, video surveillance has demonstrated its importance and benefits many times by providing immediate supervision of possessions, people, the environment and property [1]. The demand for video surveillance systems is increasing rapidly nowadays. One of the first things important to people is about their monitoring system, whether or not they have the ability to connect to it via the Internet for remote viewing. In the past, security systems had to be monitored by a guard who was locked in a room all day to monitor the video feeds to make sure nothing happened [2]. The other option was to come back and view the video footages, but at that instance, damage might be caused to the stored images under monitoring. Therefore, researchers and scientists have had to find ways to overcome this problem and thereby improve overall safety.

Simply, Surveillance can be defined as the process of closely monitoring changing information like activities, conducts, behaviors for the purpose of protecting, managing, influencing and directing people [3]. The word surveillance is derived from two French words; 'sur' which means "over" and 'veiller' meaning "watch" which is in turn derived from the Latin word 'vigilare' and is in contrast to more recent developments such as sousveillance [4]. Surveillance could also means watching over from a distance by means of electronic equipment such as Close Circuit Television (CCTV) cameras [5]. Surveillance is very helpful to law enforcement agents to investigate and prevent criminal activities, for recognizing and monitoring possible threats to an infrastructure. Basically, the design of a surveillance system consists of analyzing the needs of the people, reviewing the system costs based on existing hardware and software technologies, monitoring choices and then planning for the installation [6]. The design choices may vary from time to time due to advancement in technology. Surveillance system has always been playing a vital role in dealing with the burglary cases and protection of entire perimeter of an infrastructure [7].

Moving object detection is a critical task for many computer vision applications: the objective is the classification of the pixels in the video sequence into either foreground or background [10]. A commonly used technique to achieve this in scenes captured by a static camera is Background Subtraction (BGS) [11]. Background subtraction is the process of identifying moving objects from the portion of a video frame that differs significantly from a background scene [12].An Adaptive Gaussian Mixture Technique (AGMT) can be used to detect motion or intruders by alerting the user of possible intrusion of infrastructure under surveillance.

In this contribution, we develop a smart surveillance system using AGMT to improve the basic BGS technique in motion detection. The problem of the BGS techniques are: inability to adapt to illumination changes, inability to mask areas where motion detection is not needed like rotation of fans in homes, surveillance being placed outdoors and the existence of trees moving under the wind which affect surveillance focus [8].

Also, this technique was implemented to avoid the use of motion detection sensors like the passive infrared sensors due to the following drawbacks: high sensitivity which leads to the detection of objects such as rats, improper function in hot areas due to faulty readings caused by an increase in temperature, high rate of maintenance in which a lack of could lead to sensor failure, and high cost of implementation [9]. This work makes use of camera for both video surveillance and motion detection simultaneously. In essence, this system improves on the basic background

Article History:
Received Date: May. 10, 2020, Accepted Date: Jul. 08, 2020, Available Online: Sep. 01, 2020

subtraction technique by adapting to illumination changes in the area of surveillance and masking out area that is observed to be constantly changing, to increase accuracy of the system by making use of the adaptive Gaussian Mixture Technique in background modelling of the background subtraction algorithm

II. Review of Related Works

Several works in the area of motion detection for smart surveillance exist in: [13], [14], [15], [16], [17], [18], [8], [13] and [7]. The major limitation of these works is that they have an inability to adapt to illumination changes. Furthermore, these works require a static background for effective operation. They do not cater for situations where the foreground and background move. Due to these salient limitations, this work improves on the existing works by making use of an Application Programming Interface (API) for the remote communication and also additional image processing technique to tackle for adaptation to both foreground and background changes by applying the proposed adaptive Gaussian Mixture model in improving motion detection.

III. Methodology

This section describes the procedures used in the evelopment of the proposed system. It provides the detailed steps in block diagram, circuit diagram, flow design and flow chart. This research adopted adaptive Gaussian mixture technique in the background modelling for background subtraction technique and for image processing, as well as Twilio API for Short Message Service (SMS) notification. The adaptive gaussian mixture technique is employed to improve the performance of the conventional background subtraction technique in motion detection. The system tackles the issue of false alarm by sending SMS notification to the user on situations the user is currently not online and also for verification purposes sends the snapshots of intruder to the user's email

A. Background Subtraction Technique

Frame differencing (FD) is the most basic techniques or procedures in BGS. The technique involves finding the absolute difference in between the frame and the previous image background or frame [13]. The absolute difference is then compared with the corresponding threshold value, A, to detect objects as shown in equation (1):

$$Fg_i(x,y) = \begin{cases} 1 & |F_i(x,y) - B_i(x,y)| > A \\ 0 & Otherwise \end{cases} \quad (1)$$

Where F_i is the current frame intensity value, B_i is the background intensity value. This technique uses the background frame for all video sequences. Figure 1 shows the basic flowchart for the background subtraction technique, each successive frame in the video stream is subtracted from the modelled master background compared with the threshold to determine if foreground is detected.

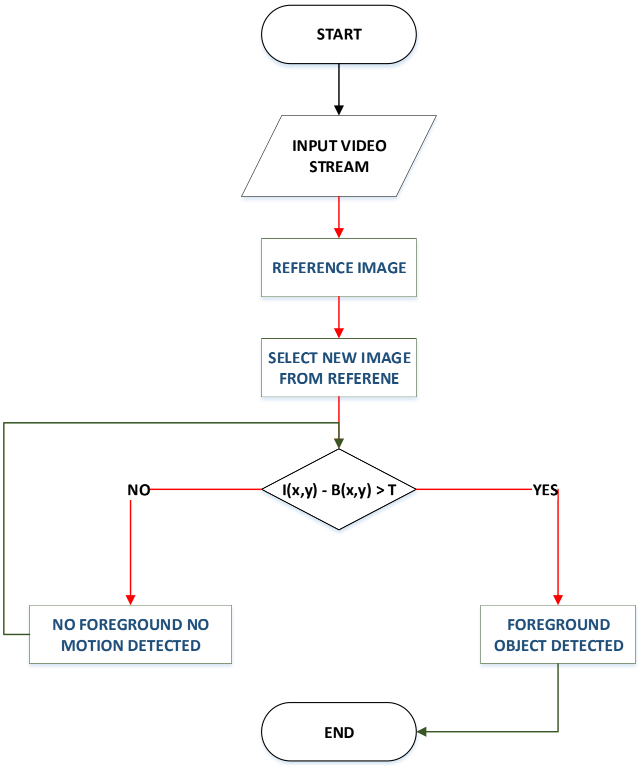


Figure 1: Basic Background Subtraction [10]

B. Mathematical Model – Adaptive Gaussian Mixture Model

In practice, the illumination in the scene could change gradually (weather conditions, or daytime in an outdoor scene) or suddenly switching light in an indoor scene like the home. A new object might be brought into the scene or an object that is present removed from it. In other to adapt to changes the model or training set (containing the initial image) is updated by adding a new sample and discarding the old one [21]. A reasonable time period T is chosen and at time t we have $= \{x^{(t)}, \ldots\ldots, x^{t-T}\}$. For each new sample we update the model X_T and re-estimate $\hat{p}$ $(\vec{x}|X_T, BG)$. However, among the samples from the recent history there could be some values that belongs to the foreground objects and we should denote these estimates as $\hat{p}\ (\overrightarrow{x^{(t)}}|X_T, BG + FG)$. We use the GMM with M components:

$$\hat{p}(\vec{x}|X_T, BG + FG) = \sum_{m=1}^{M} \hat{\pi}_m N(\vec{x}, \hat{\vec{\mu}}_m, \hat{\sigma}_m^2 I) \quad (2)$$

Where $\hat{\vec{\mu}}_1, \ldots\ldots, \hat{\vec{\mu}}_M$ are the estimates of the mean and $\hat{\vec{\sigma}}_1, \ldots\ldots, \hat{\vec{\sigma}}_M$ are the estimates of the variances that describes the Gaussian component. The covariance matrices are assumed to be diagonal and the identity matrix I has proper dimensions [22]. The Mixing weights denoted by $\hat{\pi}_m$ are non-negative and add up to one. Given a new data sample (new image model) $\overrightarrow{x^{(t)}}$ at time *t* the recursive update equations are:

$$\hat{\pi}_m \leftarrow \hat{\pi}_m + \alpha\left(o_m^{(t)} - \hat{\pi}_m\right) \quad (3)$$

$$\hat{\vec{\mu}}_m \leftarrow \hat{\vec{\mu}}_m + o_m^{(t)}\left(\alpha/\hat{\pi}_m\right)\hat{\vec{\delta}}_m \quad (4)$$

$$\hat{\sigma}^2{}_m \leftarrow \hat{\sigma}^2{}_m + o_m^{(t)}\left(\alpha/\hat{\pi}_m\right)\left(\hat{\vec{\delta}}^T{}_m\hat{\vec{\delta}}_m - \hat{\sigma}^2{}_m\right) \quad (5)$$

Where $\hat{\vec{\delta}}_m = \vec{x}^{(t)} - \hat{\vec{\mu}}_m$. Instead of time interval T that was mentioned above, here a constant α describes an exponentially decaying envelop that is used to limit the influence of data. Keeping the same notation in mind that approximately $\alpha = 1/T$. For a new sample the ownership $o_m{}^{(t)}$ is set to 1 for the 'close' component with the largest $\hat{\pi}_m$ and the others set to zero. We define that a sample is 'close' to a component if mahalanobis distance from the component is for example less than the three standard deviations [21]. The squared distance from the m-th component is calculated as follows:

$$D_m^2\left(\vec{x}^{(t)}\right) = \vec{\delta}_m^T\vec{\delta}_m / \hat{\sigma}_m^2 \quad (6)$$

If there are no 'close' components a new component is generated with $\hat{\pi}_{M+1} = \alpha, \hat{\vec{\mu}}_{M+1} = \vec{x}^{(t)}$ and $\hat{\vec{\sigma}}_{M+1} = \sigma_0$ where σ_0 is some appropriate initial variance. If the maximum number of components is reached, discarding the component with $\hat{\pi}_m$.

The presented algorithm presents an on-line clustering algorithm, usually the intruding foreground objects will be represented by some additional clusters with small weights $\hat{\pi}_m$. Therefore, we can approximate the background model by the first largest *B* clusters:

$$p(\vec{x}|X_T, BG) \sim \sum_{m=1}^{B} \hat{\pi}_m N(\vec{x}, \hat{\vec{\mu}}_m, \sigma_m^2 I) \quad (7)$$

If the components are sorted to have descending weights $\hat{\pi}_m$ we have:

$$B = arg\min_b\left(\sum_{m=1}^{B} \hat{\pi}_m > (1 - c_f)\right) \quad (8)$$

Where c_f is a measure of the maximum portion of the data that can belong to the foreground objects without influencing the background model, for example, if a new object comes into a scene and remains static for some time it will probably generate and additional stabile cluster

Since the old background is occluded, the weight π_{B+1} of the new cluster will be constantly increasing. If the object remains static enough, its weight becomes larger than c_f and it can be considered to be part of the background. Considering equation (7), It can be concluded that the object should be static for approximately $\frac{\log(1-C_f)}{\log(1-\alpha)}$ frames. For example: given $c_f = 0.01$ and $\alpha = 0.001$ we get 105 frames.

C. Motion Detection – Adaptive Gaussian Mixture Technique

Figure 2 depicts the flow diagram of the surveillance system with the proposed technique and the motion detection algorithm adopted in the system design. The initial frame is saved (modeling of the foreground) and converted to a gray scale image and then to a gaussian blur image (preprocessing stage). This is done for increasing efficiency and accuracy of result in this algorithm.

The frames with the objects is also taken and also converted to a gray scale image and then to a blur images just like the initial frame, since the first frames is already stored then the difference is gotten from the first and the second frame, once a value is gotten it can be known that there is a change in the image (movement of motion), but the object might be too small, that when captured or recorded as motion detection results to increase in false detection rate, as such a threshold or size of the picture which is calculated using the mahalanobis distance to eliminate this error readings, such that illumination changes, shadow effects and other noise effects would be reduced and its effect less.

On illumination changes it updates the initial image to the new model image to be used to calculate the difference (reducing false positives), then once the threshold is satisfied the borders of the object would be defined by adding rectangular boxes around the object. The object is recorded or image captured and saved as motion detected.

IEEE Office - SEET Complex, Federal University of Technology, Owerri, Nigeria, https://aeuso.org
EISSN: 2305-0543, PISSN: 2411-6173

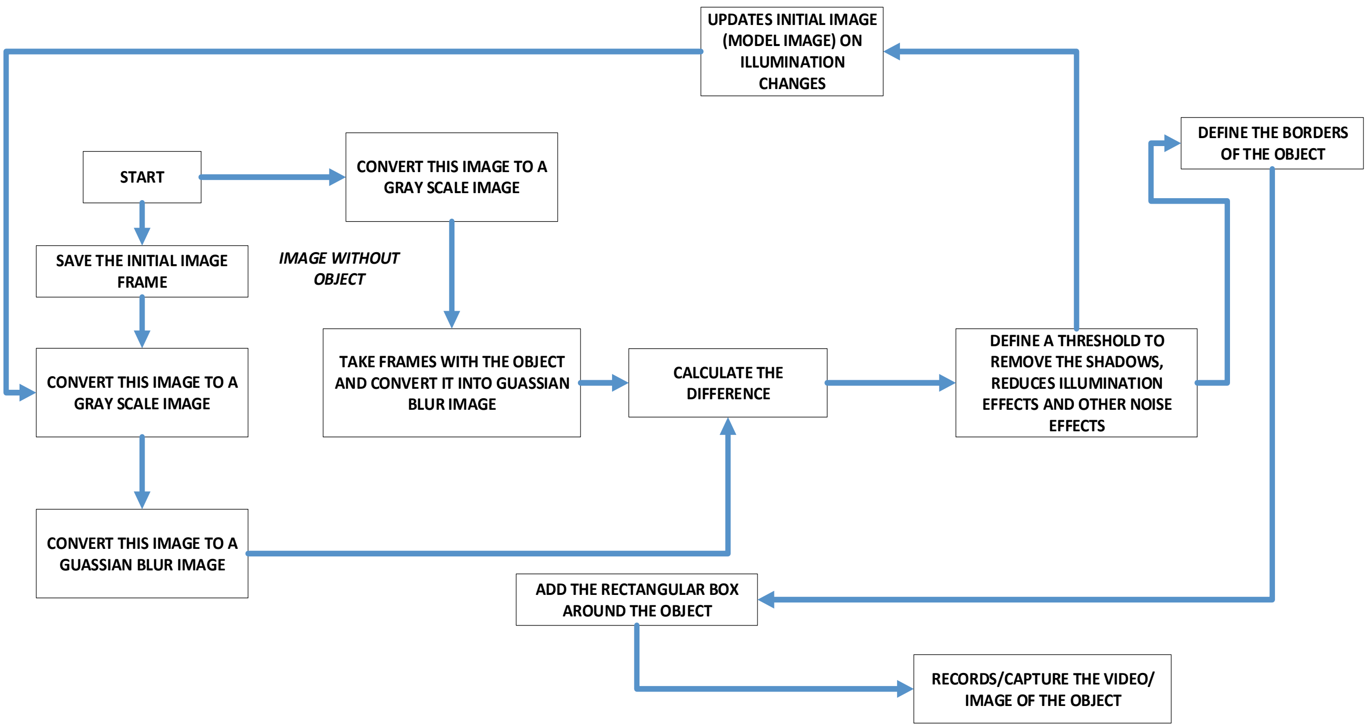


Figure 2: Flow Diagram of the Developed Enhanced Background Subtraction Technique (AGMT)

D. Algorithm – Adaptive Gaussian Mixture Technique

Learning:

Input: δ, β, x

$K > 0, \boldsymbol{\sigma_{ini}^{-1}} = (\delta std(\boldsymbol{X}))^{-1}, M = \theta$

For all input data vector $\boldsymbol{x} \epsilon \boldsymbol{X}$ do

If $K = 0$ or $\exists j, d_M^2(x,j) < X_{D,1-\beta}^2$

then

Update(x)

Else

$M \leftarrow M \cup create\ (x)$

End if

End for

Update:

Input: x

For all Gaussian components $j \in M$ do

$$d_M^2(x,j) = \left(x - \mu_j\right)^T \boldsymbol{\Lambda_j}(x - \boldsymbol{\mu_j})$$

$$p(x|j) = \frac{1}{(2\pi)^{D/2}\sqrt{|C_f|}} \exp(-\frac{1}{2} d_M^2(x,j))$$

$$v_j(t) = v_j(t-1) + 1$$

$$sp_j(t) = sp_j(t-1) + p(j|x)$$

$$e_j = x - \mu_j(t-1)$$

$$w_j = \frac{p(j|x)}{sp_j}$$

$$\Delta\mu_j = w_j e_j$$

$$\Delta\mu_j = \mu_j(t-1) + \Delta\mu_j$$

$$e_j^* = x - \mu_j(t)$$

$$\bar{\Lambda}(t) = \frac{\Lambda(t-1)}{1-w} - \frac{\frac{w}{(1-w)^2}\Lambda(t-1)e^* e^{*T}\Lambda(t-1)}{1 + \frac{w}{1-w} e^{*T}\Lambda(t-1)e^*}$$

End for

Create:

Input: x

$K \leftarrow K + 1$

return new Gaussian component K with $\mu_k = x, \Lambda_k = \sigma_{ini}^{-1} I, |C_k| = |\Lambda_k|^{-1}, sp_j = 1, v_j = 1, p(j) = \frac{1}{\sum_{k=1}^{K} sp_j}$

The system made use of three basic components in the embedded system design as underlisted:

- Raspberry Pi Model 3B
- Raspberry Pi 5mp Camera module

Figure 3 shows the circuit diagram of the system integration, the camera module is inserted into the camera port of the raspberry pi board through the camera cable and the raspberry pi is also connected to the power source, the proposed algorithm is implemented in the raspberry pi using python programming language to perform the motion detection, sending SMS notification and also sending

IEEE Office - SEET Complex, Federal University of Technology, Owerri, Nigeria, https://aeuso.org
EISSN: 2305-0543, PISSN: 2411-6173

of the video footage or captured images of intrusion to cloud to be accessed by the user.

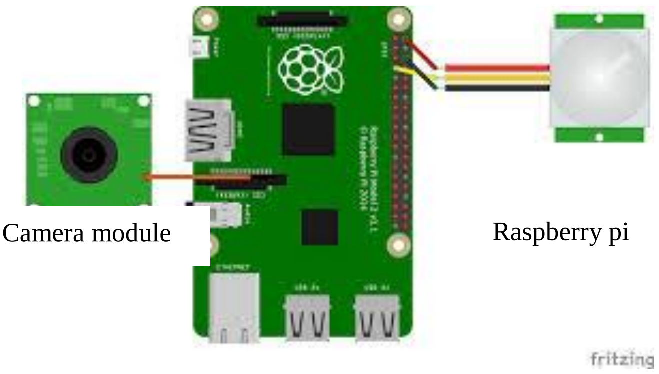


Figure 3: Circuit Diagram showing the Connection of the Raspberry pi

The SDcard is loaded with the Raspbian operating system and inserted into the raspberry pi 3 board and booted, the OpenCV is installed into the raspberry pi to enable the image processing algorithm to be implemented in this case motion detection. The motion detection and SMS notification script are created, and stored in the home of the raspberry pi to be loaded on boot. This SMS script links to the API supplied by Twillo for SMS notification to the registered number as such the user would be able to login and view video footage or captured images.

Figure 4 depicts the dataflow diagram of the smart surveillance system developed by applying the adaptive gaussian mixture model in motion detection.

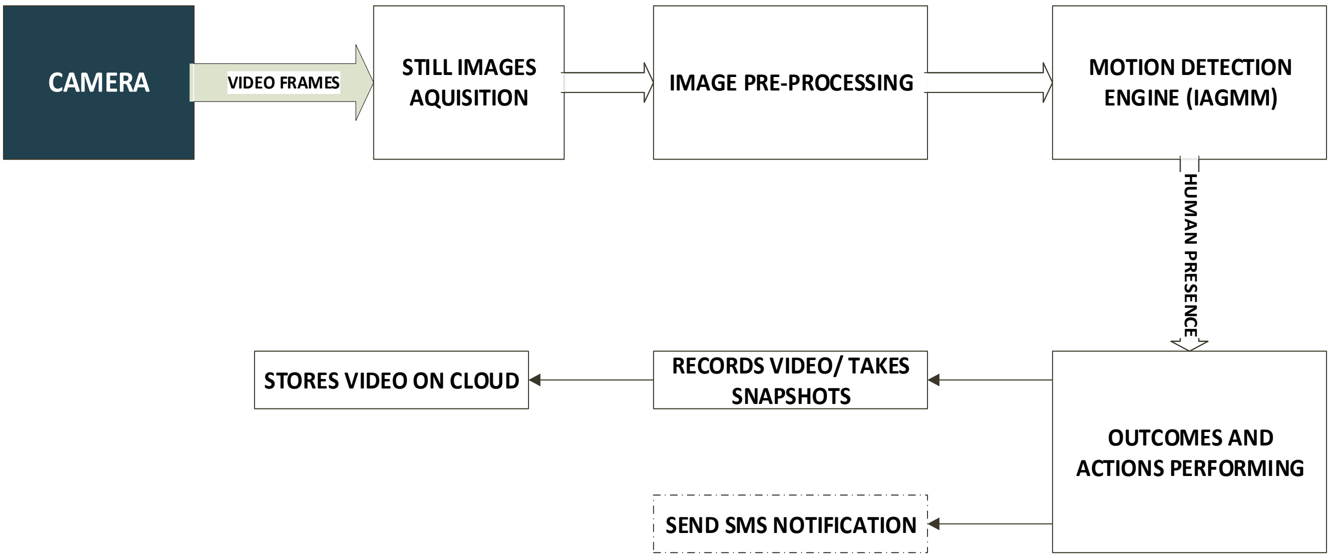


Figure 4: Data Flow Diagram of the Developed System

E. System Working Principle

The electronic components used for the smart surveillance system development are, Raspberry Pi 3, Pi-Camera module, GSM module, Switch button, power supply. The camera, the push button and the power supply acted as the input to the microcontroller (Raspberry P3) while the alert system and GSM module acts as the output for the system, the API (application interface) acts as both input and output in the surveillance system. Once images/frames are being gotten from the camera into the Raspberry pi3 controller, the image is being processed using the enhancement techniques by applying the adaptive Gaussian Mixture model to model the background for background subtraction, the images that undergo background subtraction is stored within the programmed Raspberry pi3. The system operation flowchart is shown in Figure 5.

IEEE Office - SEET Complex, Federal University of Technology, Owerri, Nigeria, https://aeuso.org
EISSN: 2305-0543, PISSN: 2411-6173

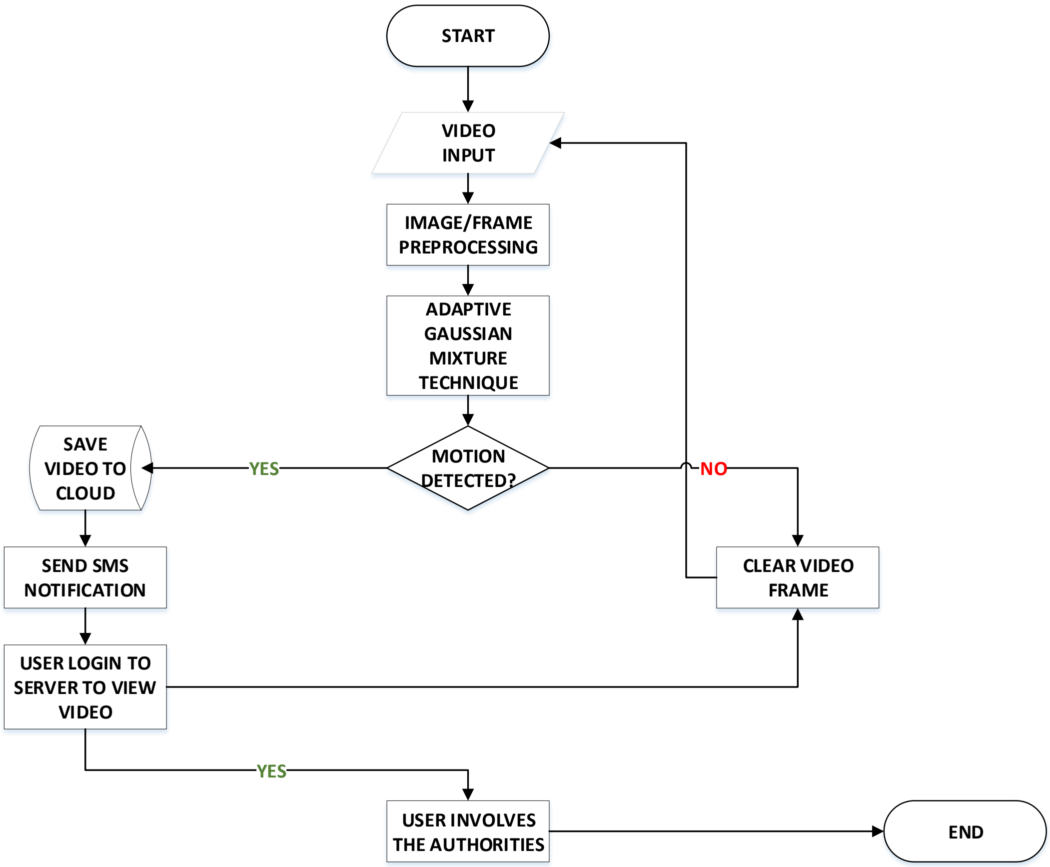


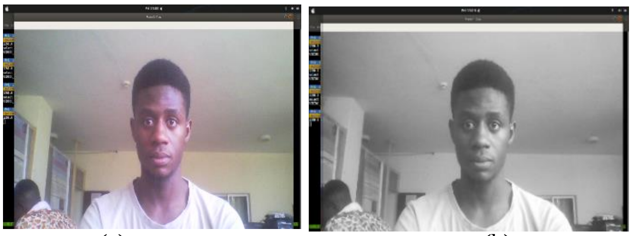
Figure 5: Flow Chart Diagram of the Developed System

IV. Results and Discussion

This section contains results obtained and discussion based on the result. The simulation of the system was accomplished using OpenCV and using the Webcam of the laptop, to get results on the implemented motion detection algorithm. Also results from the integration of the software and hardware parts of the system are reported. Also, the false positives rates and true positives rates obtained are plotted on the Receiver Operating Characteristics (ROC) to show the level of accuracy for the implemented algorithm.

A. Simulation Results – Motion Detection

Figure 6(a) shows the raw image obtained from the webcam in RGB values and each has 3 bytes per pixel which is considerably large for the computer system to process. The image is then converted first to gray image as shown in Figure 6(b), this gray image has just a single byte per pixel as such reduces the size of the image to be processed but during motion, movement from one frame to other results to creating rough edges. The image is further converted to a blur image to make the edges smoother so the frame changes would not be noticed and also to increase the efficiency of the algorithm as shown in Figure 7.



(a) (b)

Figure 6: Frame Inputs from Webcam

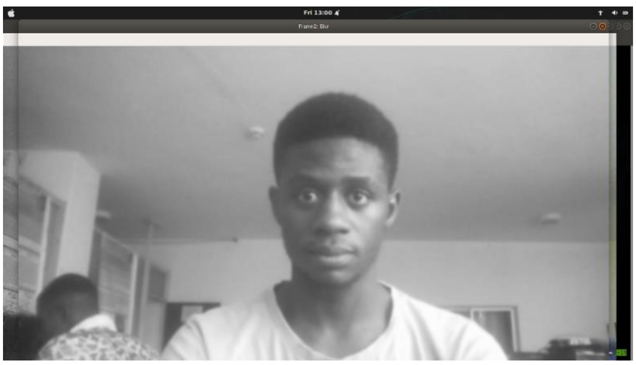
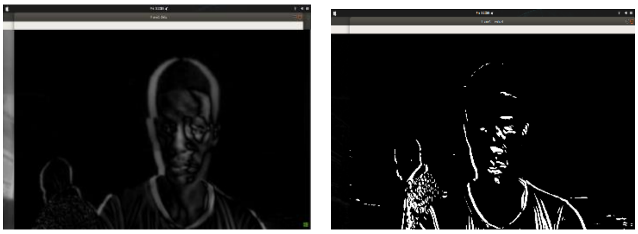


Figure 7: Blur Frame Input

IEEE Office - SEET Complex, Federal University of Technology, Owerri, Nigeria, https://aeuso.org
EISSN: 2305-0543, PISSN: 2411-6173

The algorithm processes every image frame and the motion detection algorithm finds the difference in subsequent image frames as shown in the delta result in Figure 8(a). This delta frame shows only the part between frames that are changing after which a threshold is applied to the delta image resulting in the threshold frame in Figure 8(b).



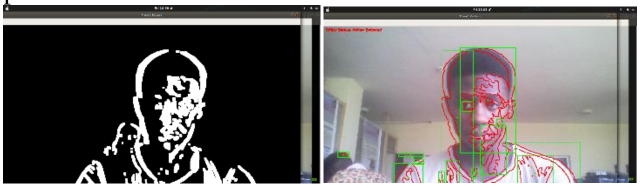
(a) Delta Frame (b) Threshold Frame

Figure 8: Image Segmentation

From figure 8(a), background subtraction has been achieved by subtracting subsequent frames from the master frame this resulting frame is known as the delta frame showing some gray parts in the frame. In Figure 8(b), Threshold is applied to the delta frame and according to the algorithm every pixel value that is above 15 would go to 255 and becomes white and if it is 15 or below it goes to zero and becomes black as such it changes everything from white to black. Thresholding is very critical to motion detection as the value determines how sensitive or accurate the algorithm is.

Having very small threshold would increase false detection rate, as such the system performance reduces and the receiver operating characteristics as shown in the performance evaluation section would approach the diagonal shape, very little accuracy, likewise having a very large threshold value, so a balance have to be made for optimal results. Figure 9(a) shows the dilated frame this frame simply expands what shape is gotten from the threshold frame; all this are functions in OpenCV. In Figure 9(a) every shape is being expanded in the dilated frame or blob, a contour is any shape that is built around a blob where everything is the same color.

In this case having just black and white, it becomes very easy to accomplish as such a contour is built around the white blob, and it is identified as the red boundary color shown in Figure 9(b). The centroid of the contours is signified by the red dot, the green box is just a bounding box around the contours or object being detected and the centroid of the rectangle is represented by the green dot. The threshold value can be set to be extremely sensitive or less sensitive, in this simulation result a threshold of 15 was used, but in implementation on the hardware to reduce likelihood of detecting very little change the threshold was increased to about 25, and the master frame updated regularly to account or tackle illumination problems.

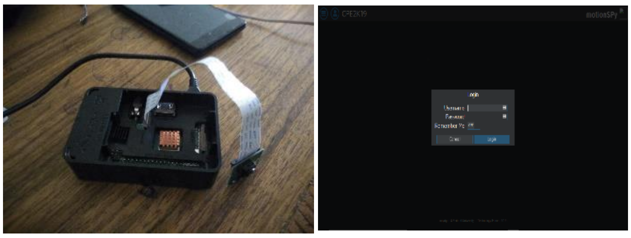


(a) Dilated Frame (b) Contour Frame

Figure 9: Motion detection

B. Hardware Implementation

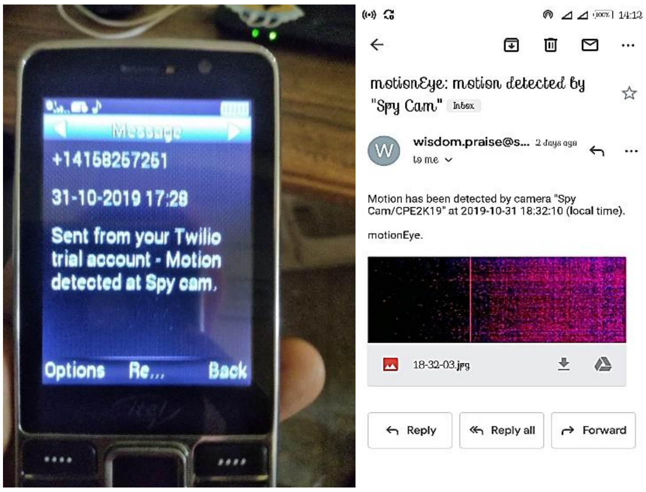
Figure 10(a) shows the hardware integration of components, the Raspberry Pi 3 and Pi camera module. The system is accessed through a web platform where the user can view the video footage of the home being monitored, as shown in Figure 10(b) and also an admin page where some settings changes can be made to improve the system. The web application was designed using Django web framework which is a python-based backend framework, the motion detection script activates on turning on the system.



a. Hardware assembly b. Web interface

Figure 10. Hardware and web interface

Figure 11(a) shows the SMS notification received on the phone number registered in the Twilio account on motion detected by the Spy cam. Figure 11(b) shows email notification sent to the registered user's email including snapshot of the intruder that can be downloaded for further investigation and sent to the investigating authority. Also, for further proof the system records a 10 seconds movie and stores it on the raspberry server and can be accessed and viewed by the user.



a. SMS notification b. Email Notification

Figure 11. Alert notification

C. *Performance Evaluation*

The system was evaluated using sensitivity and specificity performance evaluation metrics. The Sensitivity metric (also called true positive rate, given, or the probability of detection in some areas) measures the proportion of actual positives which are correctly identified as such (for example, the percentage of sick people who are correctly identified as having the condition).

On the other hand, Specificity metric (also called true negative rate) measures the proportion of actual negatives that are correctly identified as such (for example, the percentage of healthy people who are correctly identified as not having the condition). In the terminology true or false positive or negative, true or false refers to the assigned classification is correct or incorrect, while positive or negative refers to the allocation to the positive or the negative category.

$$sensitivity = \frac{number\ of\ true\ positives}{number\ of\ true\ positive + number\ of\ false\ negatives} \quad (9)$$

$$sensitivity = \frac{number\ of\ true\ positives}{total\ number\ moving\ objects} \quad (10)$$

$$sensitivity = probability\ of\ motion\ detection\ given\ that\ motion\ occured \quad (11)$$

Specificity refers to the system's ability to correctly not detect motion provided motion or intruder movement did not occurs in the video stream. Mathematically specificity is given as:

$$specificity = \frac{number\ of\ true\ negatives}{number\ of\ true\ negatives + number\ of\ false\ positives} \quad (12)$$

$$specificity = \frac{number\ of\ true\ negatives}{total\ number\ of\ nonmoving\ objects} \quad (13)$$

$$specificity = probability\ of\ not\ detecting\ motion\ given\ that\ motion\ did\ not\ occur \quad (14)$$

For different threshold values in this work, tested for 7 threshold values the sensitivity was measured in Table 1:

Table 1: Showing sensitivity at various thresholds

S/N	Thresholds (Pixels)	Sensitivity (%)	Specificity (%)
1.	5	90	10
2.	10	88.5	10
3.	15	80	92
4.	25	77.9	86.5
5.	27	79.3	90
6.	30	40.5	99
7	50	33.3	100

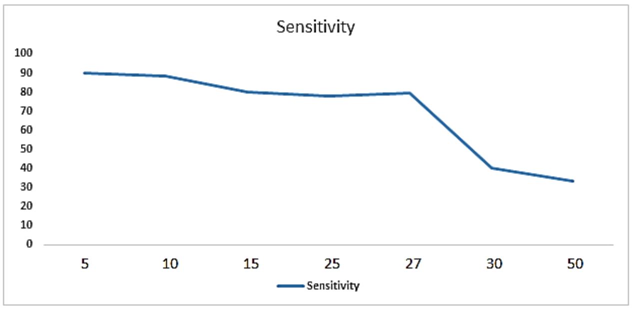


Figure 12: Sensitivity graph with respect to threshold

The ROC curve is created by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR) to different threshold settings. The true positive rate is also known as the sensitivity, recall or the probability of motion detection in the algorithm. The false positive rate is also known as the fallout or the probability of false alarm and can be calculated as (1 - specificity). It can also be considered a plot of power as a function of the type I error of the decision rule (when the performance was calculated from only a sample of the population, it may be considered estimators of these quantities). The ROC curve is thus the sensitivity as a function of fall-out.

Table 2: Table relating Sensitivity to fallout

S/N	Thresholds (Pixels)	Sensitivity (%)	Fallout = (100-Specificity) (%)
1.	5	90	90
2.	10	88.5	90
3.	15	80	8
4.	25	77.9	13.5
5.	27	79.3	10
6.	30	40.5	1
7	50	33.3	0

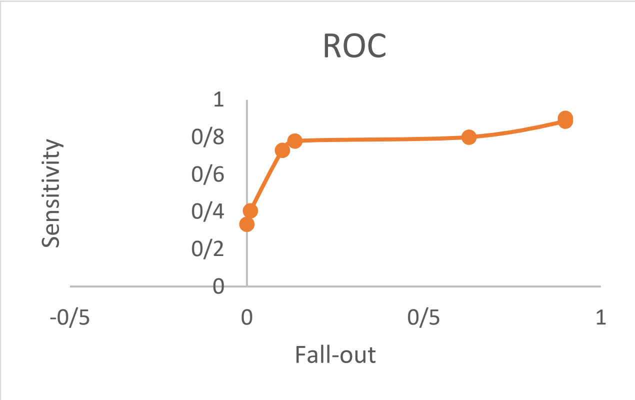


Figure 13: Receiver Operating Characteristics curve

From the ROC analysis, the closer a contingency table is to the upper left corner of the curve, the better it detects as such looking at the graph the best threshold to satisfy this condition is the threshold of 15 as it approaches the top left corner of the graph in Figure 13 in relation to Table 2.

D. Basic Background Subtraction Technique with Adaptive Gaussian Mixture Model

The basic background subtraction technique algorithm was simulated, the sensitivity and specificity were recorded in Table 3. the sensitivity and specificity at the threshold of 15 represented in the bounding box in Table 3 compared to Table 2 was optimized and improved on adopting the adaptive Gaussian Mixture Technique in Background Modelling.

This result shows that the implemented enhanced background subtraction technique improved on the basic background subtraction by reducing false positives in motion detection. From Table 3. the sensitivity and specificity at the generally tested which was seven different thresholds in this study. Results shows to be relatively lower than the adopted Gaussian Mixture Technique and at threshold of 15 the basic motion detection algorithm records a sensitivity of 55.5% and a specificity of 40.5% which is very poor and yielding a lot of false positives in motion detection and susceptibility to illumination changes.

Table 3: Table relating Sensitivity to fallout

S/N	Thresholds (Pixels)	Sensitivity (%)	Specificity (%)
1.	5	99	50
2.	10	32.5	37.5
3.	15	55.5	40.5
4.	25	46.7	45.7
5.	27	40.5	47.5
6.	30	35.7	49.8
7	50	33.3	50.5

V. Conclusion and Future Work

Really, security of valuable properties against intruders should be paramount concerns to all from both local and global contexts. Various methods are already in place to tackle insecurity issue in motion surveillance systems. This paper presented an improvement to object background subtraction technique. The system developed around the Adaptive Gaussian Mixture technique allowed detection of intrusion. Provide notification via SMS for low internet service penetration like Nigeria. In other words, the system is remotely controlled by the user who is not within the vicinity. The developed system is initiated to tackle a fragment of a much broader problem by focusing on formal areas such as homes and small office premise.

Future improvements could be to in the aspect of motion discrimination of human intruders and household pets. Also, this system can be extended to include PIR sensors to improve system performance and accuracy, hence, this could be applied using the redundancy approach to optimize the system in tackling intrusion and reducing criminal activities.

VI. References

[1] S. Ackerman and S. Thielman, US intelligence chief: we might use the internet of things to spy on you, US: The Guardian, 2016.

[2] Fujika, Plakninly and James, The Surveillance Studies Centre, Queen's University: A multi-disciplinary research centre at Queen's University, 2017.

[3] D. Lyon, Surveillance Studies, Cambridge: Polity Press, 2007.

[4] K. R. M. S. Minsky M, The Society of Intelligent Veillance, Tornto, Otario, Canada: Proceedings of the IEEE ISTAS 2013, 2013.

[5] Leighton, W. Kille, Maximino and Martin, The effect of CCTV on public safety: Research roundup, Harvard: Harvard Kennedy School's Shorenstein Center and the Carnegie-Knight Initiative, 2014.

[6] S. Singhd, IP Camera Video Surveillance using Raspberry Pi.,, 2015.

[7] N. b. Harum, M. F. Ali, N. A. Zakaria and S. Anawar, "Smart Surveillance System using Background Subtraction Technique in IoT Application," *Centre for Advanced Computing (C-ACT), Faculty of Information and Communication Technology,* vol. IV, no. 12, 2018.

[8] M. F. Chauhan and G. Anuradha, "Design and Develop Real Time Video Surveillance System Based on Embedded Web Server Raspberry Pi B+ Board," *International Journal of advanced Engiineering and Research Development (ijaerd),* pp. 1-4, 2015.

[9] N. Yang, "Motion sensor and camera placement design for in-home wireless video monitoring systems," *IEEE Globalcom,* 2011.

[10] k. Ashish and A. Choubaey, "Motion Detection Surveillance System using background subtraction algorithm," *Advance Research in Computer Science and management studies,* 2013.

[11] S. Herrero and J. Bescós, "Background Subtraction Techniques: Systematic Evaluation and Comparative Analysis," *ACIVS '09 Proceedings of the 11th International Conference on Advanced Concepts for Intelligent Vision Systems,* pp. 33-42, 2009.

[12] S.-C. Cheung and C. Kamath, "Robust Background Subtraction with Foreground Validation for Urban Traffic Video,," *EURASIP Journal on Applied Signal, Volume 14,,* pp. 1-11, 2005.

[13] k. Ashish, A. Choubaey, A. Michin and I. Funkar, "Motion Detection Surveillance System using an improved background subtraction algorithm," *Advance Research in Computer Science and management studies,* 2019.

[14] C. Sureindar and D. P. Saravanan, "Motion Detection using optical Flow on Raspberry Pi," *International Research Journal of Engineering and Technology (IRJET),* vol. II, no. 8, November 2015.

[15] P. A. Shake and S. S. Borde, Embedded surveillance system using PIR sensor, vol. II, 2014.

[16] G. Sebastian, V. H. Peter and W. Philips, "An Edge-based Appraoch for Robust Foreground Detection," Ghent University TELIN-IPI-IBBT, Sint Pietersnieuwstraat, Gent, Belgium, 2013.

[17] S. Madhavi, S. Rutuja, T. Neelam and B. Amol, "Security Based Home SUrveillance System Using android application," *International Journal of Research in Engineering and Technology,* 2014.

[18] Jadhav, Real Time Embedded Network Video Capture And SMS Alerting system,, 2014.

[19] P. Sanjana, J. S. Clement and S. Rhedeer, Smart surveillance system using raspberry pi and PIR sensor, 2014.

[20] K. U, M. R, S. S and P. A, "implementation of a low cost wireless image Acquisition And Transfer To Web Client Using Raspberry Pi For Remote Monitoring," *International Journal of Computer Networking, Wireless and Mobile Communications (IJCNWMC),* vol. IV, pp. 17-20, 2017.

[21] S.-T. Su and Y.-Y. Chen, "Moving Object Segmentation Using Improved Running Gaussian Average Background Model," *Digital Image Computing: Techniques and Applications,* 2017.

[22] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," The Artificial Intelligence Laboratory MIT, Cambridge,MA 02139, 2015.

IEEE Office - SEET Complex, Federal University of Technology, Owerri, Nigeria, https://aeuso.org
EISSN: 2305-0543, PISSN: 2411-6173