Journal of Harbin Engineering University Vol 45 No. 4
ISSN: 1006-7043 April 2024

Comparative Study of Background Subtraction Algorithm for Moving Object Detection using OpenCV

[1]Ms. Vinaya Kulkarni , [2]Mrs. Tejaswee Pol , [3]Dr. Shrikant Mapari
[1]SIES College of Arts , Science and Commerce (Empowered Autonomous), Sion(West), Maharashtra, India.
[2]Ramniranjan Jhunjhunwala College of Arts, Science and Commerce (Empowered Autonomous Status), Ghatkopar(West), Maharashtra, India.
[3]Symbiosis Institute of Computer Studies and Research (SICSR),Symbiosis International (Deemed University) (SIU),Model Colony, Pune, Maharashtra, India.

Abstract
The object detection involves detecting objects in images and videos using computer vision, image processing, and deep learning. OpenCV is a cross-platform library for developing real-time computer vision applications. In addition to image processing and video capture, it includes face and object detection features. This paper proposes a background subtraction algorithm for detecting moving objects. Video sequences are detected using background subtraction methods. Several parameters of the video sequence could complicate this process like noise, wind, rain, etc. This paper aims to compare the Background subtraction algorithm which is a Mixture Of Gaussians – MOG, K-Nearest Neighbour (KNN), CNT (COUNT), GMG, and MOG2 on the same data set.

1. Introduction:

Background subtraction

The background subtraction method is widely used for detecting moving objects from static cameras. A basis for the approach is to detect moving objects by comparing the current frame with a reference frame, often referred to as a "background image" or a "background model". Numerous applications in computer vision use background subtraction, such as surveillance tracking and human pose estimation. In real environments, background subtraction is usually based on a static background hypothesis, which can lead to problems when variations shift in the external environment such as rain, ice, etc. These techniques can not be if the camera recording the landscape in a moving car as the scenario or background will change in each video frame.

The Following are the sequence of steps of the algorithm:

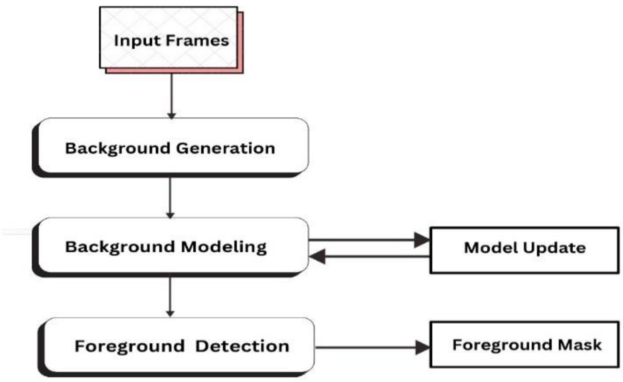


Figure 1: Flow Chart of the model

- background generation – processing N frames to provide the background image
- background modelling – defining the model for background representation
- background model update – introducing the model update algorithm for handling the changes, which occur over time
- foreground detection – dividing pixels into sets of background or foreground.

2. Background Subtraction Algorithm:

2.1 Temporal Median Filter:

The pixel intensity is calculated using the function V(x, y, t), where t is the time dimension while x and y are variables which store the pixel locations. For example, V(1, 2, 3) is the pixel intensity at the location (1, 2) of the pixel at t = 3 in the video sequence.

B(x, y, t) = $\frac{1}{N}\sum_{i=1}^{N} V(x, y, t-i)$

Where N is the number of previous images for the average calculation.

Calculate the background B (x, y, t) and subtract from the image V (x, y, t) in time t = t and limit it to the threshold (Th).

| V(x, y, t) – B(x, y, t) | > Th

For example, there are 25 background estimates if 25 frames for training are used. After the calculation, the background model is generated. At each new input image (or frame), the algorithm compares each pixel with the median value of the created model.

Steps:

1. Convert the background template to grayscale.
2. Loop all the frames of the video. Extract the current frame and convert it to grayscale.
3. Calculate the absolute difference between the current frame and the background model.
4. Apply threshold to remove noise and binarize the output (only 0 and 1 – or 0 and 255).

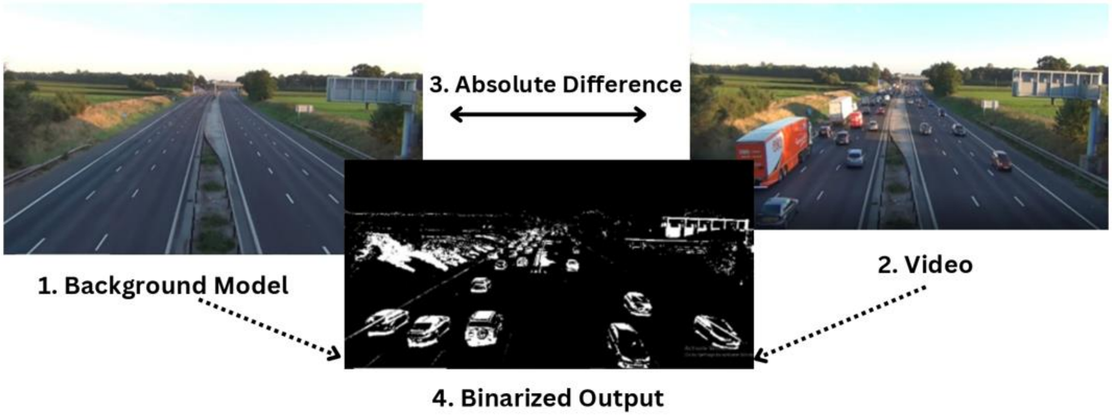


Figure 2: Background Subtraction

2.2 Mixture Of Gaussians – MOG

It is a Gaussian Mixture-based Background/Foreground Segmentation Algorithm. It was introduced in the paper "An Improved Adaptive Background Mixture Model for real-time tracking with Shadow Detection" by P. KadewTraKuPong and R. Bowden in 2001. It uses a method to model each background pixel by a mixture of K Gaussian distributions (K = 3 to 5). The weights of the mixture represent the time proportions that those colours stay in the scene. The probable background colours are the ones which stay longer and more static.[3]

It is assumed that the intensity values V (x,y,t) of each pixel in the video can be clustered by selecting an appropriate Gaussian distribution number for each pixel.

Step 1: Learn the background model

• Each pixel is characterized by its intensity V (x,y,t) in the RGB color space

• It uses multiple Gaussian distributions to model the background of a pixel. Each Gaussian takes
3 parameters:

1. Mean ($\mu_{i,t}$): estimates of the mean intensity of the colour.
2. Variance ($\sigma_{i,t}$): estimates of the variation from the mean.
3. Weight (k): number of Gaussians per pixel, representing the amount of time the colours were present in the scene.

$\sum_{i=1}^{K} \omega_{i,t}$. η(u; $\mu_{i,t}$, $\sigma_{i,t}$)

• Step 2: Classifying the pixels

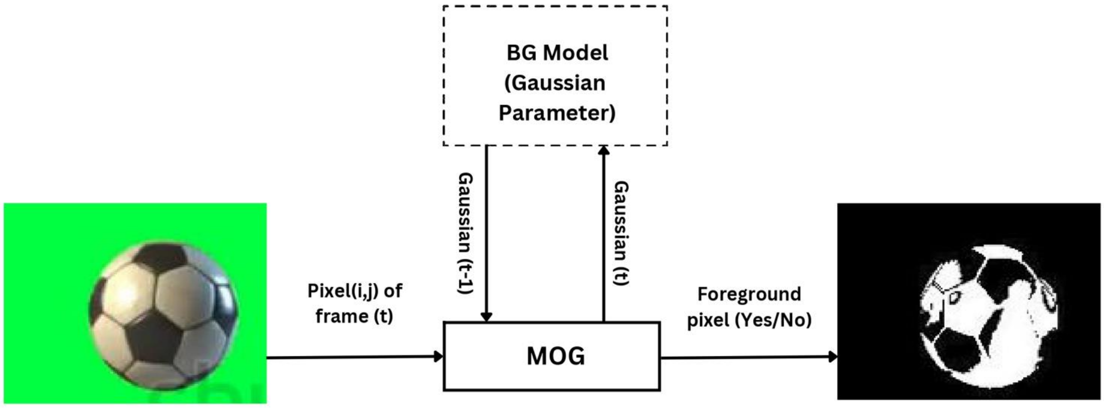


Figure 3: Flow Chart for MOG algorithm

For each new pixel, the Gaussian parameters are updated individually (independent of other pixels) based on the learning rate to fit the scene. It takes into account the value of the new input pixel in order to track background changes.

• Pixels that do not match the "background gaussians" are classified as foreground

• Foreground pixels are clustered if the average of the Gaussians matches the new pixel value.

2.3 GODBEHERE, MATSUKAWA, GOLDBERG (GMG)

The name of the algorithm GMG is derived from the initials of the inventors who proposed the algorithm. It uses Bayesian inference to separate the background from the foreground. The algorithm combines background estimation with Bayesian segmentation per pixel. A history of frames is also used to model the background. Again, it is weighted by the time sequence of the frames. A greater weight is assigned to the new observation than the previous one.

<u>Identify possible foreground objects using Bayesian inference</u>

• For each new pixel the algorithm will classify it according to the probability given by Bayesian inference.

P (A|B) = $\frac{P(B|A) * P(A)}{P(B)}$

<u>The Kalman Filter (Linear Quadratic Estimation – LQE)</u>

- Kalman Filter (Linear Quadratic Estimation - LQE) is an algorithm that uses a series of measurements observed over time along with statistical noise. By estimating a joint probability distribution over the variables for each timeframe, it produces estimates of unknown variables that are more accurate than those based on a single measurement alone.
- It uses the first few hundred frames (120 by default, +- 5 seconds) of the input video to build the model.
- The second stage filters the pixels as a foreground to reduce the noise present in the first stage. Newer pixels are given a higher weight than older pixels to adjust the variations in lighting.

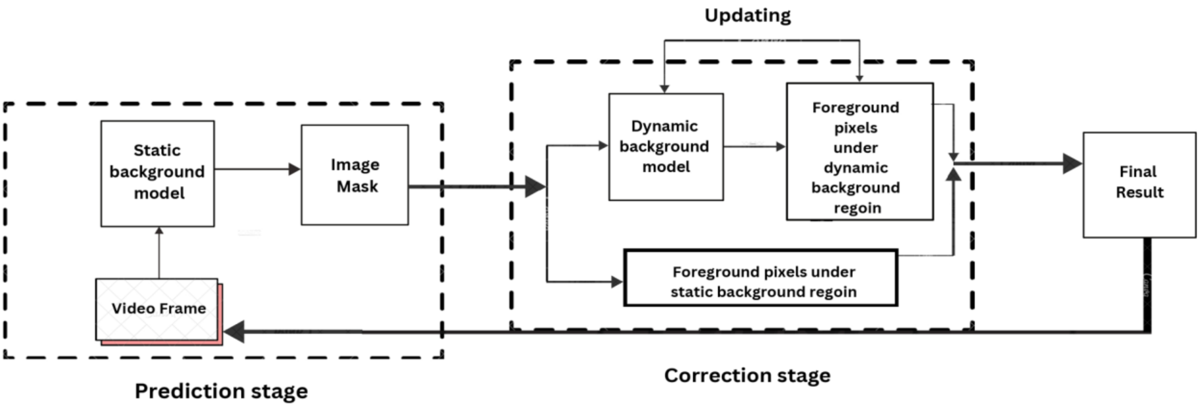


Figure 4: Flow Chart for Kalman Filter

2.4 K-NEAREST NEIGHBOR (KNN)

The K-Nearest Neighbor (KNN) algorithm is a popular machine-learning technique for classification and regression tasks. It assumes that similar data points have similar labels or values. In the training phase, the KNN algorithm uses the entire training dataset as a reference. Predictions are made by calculating the distance between the input data point and all the training examples, based on a distance metric of choice.

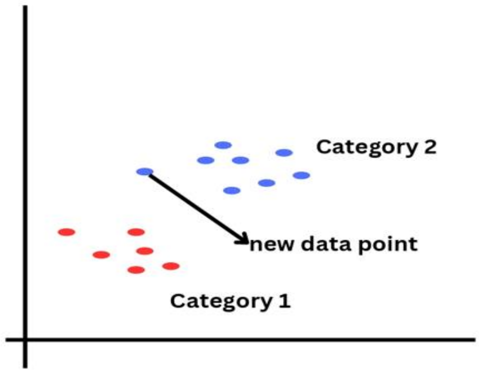


Figure 5: KNN

2.5 CNT (COUNT)

- It is a method that uses only the information of the pixel values in the previous nFrames and other additional information.
- Background subtraction based on frame count.
- If the pixel colour remains stable (PixelStability()) for a certain number of frames it is considered as background, otherwise it is "unstable" and classified as foreground.
- Much faster than any other background solution in OpenCV (optimized with the Valgrind tool).

3. Literature Review:

In this paper author (Trnovszký, Peter Sýkora and Róbert Hudec) proposed the method for background subtraction method are used to detect foreground objects in video sequences. Comparison is realized by using twenty video sequences captured in the near-infrared spectrum. Each video sequence has one or more moving wild mammals. The moving objects are manually segmented on twenty randomly selected frames for each video. A group of people does manual segmentation. Then, results from background subtraction methods are compared to human segmentation using a brute force matcher and improved by using Radon transformation. The results show that the KNN has the most significant similarity opposite to human segmentation.

In this paper authors (Hanchinamani, Sayantam and Satish) proposed a high-speed background subtraction algorithm for moving object detection is proposed. The video is first converted to streams and then applied to a convolution filter which removes high-frequency noise components to obtain smoothened images. The smoothened images are then applied to the background subtraction algorithm with an adaptive threshold that gives the detected object in the background image. The detected object is then applied to a convolution filter to remove the spurious distorted pixels, improving the image's quality. The proposed architecture was designed using VHDL language and implemented using Spartan-6 (XC6SLX45-2csg324) FPGA kit. It is observed that the proposed technique is better than the existing method in terms of image quality and speed of operations.

In this paper authors (Limbasiya and Pratik) Proposed that System Detection and Tracking are done through several steps: Background Modeling, Foreground and Feature Extraction, Object Detection, Object Modeling, and Analysis of objects. In this continuous process, For Background Modeling, there is are different algorithms like W4 (What? Where? Who? When?), Median Algorithm, and HRR (Highest Redundancy Ratio) Algorithm. For the Motion Analysis, there is different Method like: Egin Gait, Template Matching, Baseline Algorithm, and Star Skelton Model using Human Gait. The author found that HRR is best for Background Modeling with less Computational time and Star

Skeleton Model for Human Recognition with less computational Cost.

In this paper authors (Wei Liu) propose a robust framework for scene background modelling based on temporal median filter with Gaussian filtering. Specifically, each background pixel is firstly modeled with a probability density function (PDF) learned over a series of video sequence. Then, pixels in video sequence with low probabilities are filtered, taken as foreground moving objects or noises. Finally, a temporal median filter is employed on video sequence, with pixels left. The performance of our framework is evaluated visually and numerically using different metrics on the scene background modeling contest 2016.

In this paper, author (Wang, Y. and W.) aimed to recognize human actions when given acceleration signals. The author apply KNN classifier to the test set and studies the relationship between the value k in KNN and the recognition accuracy. The numerical results indicate that with an appropriate value of k, the recognition accuracy can reach 96.70%, which can meet the engineering requirements. This study not only puts forward an approach for human action recognition but also exhibits the broad prospect of machine learning methods for solving recognition problems.

In this paper, author (Zheng Yi) suggest that To detect moving objects from video sequences with a complex background, propose an algorithm based on running average background modeling and temporal difference method. Firstly, utilize the running average method to dynamically updating the background image. Through using background subtraction, we get a foreground image. Secondly, use temporal difference method to get a difference image. By combining the foreground image with the difference image, the common information between them can be achieved. Finally, we eliminate the noise in the combined image by using the median filter, and then we can get the moving objects. Experimental results show that, comparing with traditional running average method, temporal difference method and Gaussian mixture background modeling method, our method can detect the moving objects from complex backgrounds more accurately with low computational complexity.

4. Methodology

4.1Step 1: Background subtraction using the Temporal Median Filter

a. Captured 25 random frames that will be used to generate a background model.
b. Extract a median of 25 frames to generate a background model, which will be used to compare with each one of the frames of the video so it will be possible to detect the object.

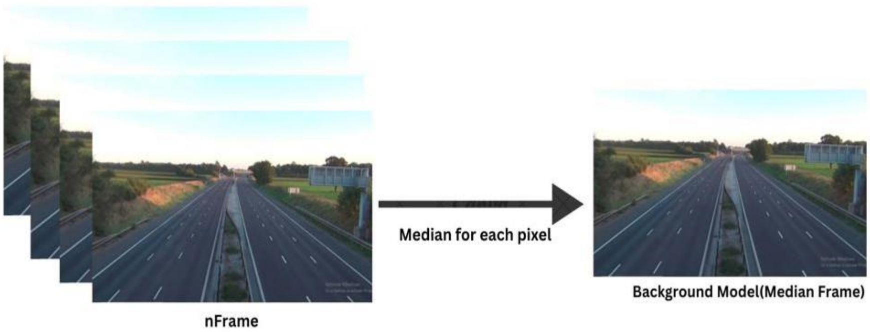


Figure 6: Background using Temporal Median Filter

c. Convert the generated background model to grayscale.
d. Convert the background model to grayscale.
e. Loop all the frames of the video. Extract the current frame and convert it to grayscale.
f. Calculate the absolute difference between the current frame and the background model.
g. Apply threshold using the OSTU threshold method to remove noise and binarize the output (only 0 and 1 – or 0 and 255). The OSTU threshold method analyses the pixel and decides the best threshold.

4.2 Step2: Background subtraction using the GMG, MOG2, MOG, KNN and CNT

a. Define a function get_kernel() to return a structuring element i.e. kernel or matrix that is used to perform multiplication. (e.g. cv2.MORPH_ELLIPSE(3,3)).

b. Define a function get_filter() to apply morphological operations i.e. Erosion, dilation, Opening, and Closing.

c. Define a function get_bgsubtractor() to implement the algorithm that are being used GMG, MOG2, MOG, KNN and CNT

d. Loop all the frames of the video. On each frame apply background subtractor algorithms (GMG, MOG2, MOG, KNN, and CNT) sequentially and compare the result. On the returned result apply morphological operations followed by bitwise and operator to improve the result. By applying a combination of techniques, the result gets better.
With opening, the result is not that good, it misses some objects in the video.
With the closing, the result is quite better than the opening.
For all algorithms, the best morphological operation is a combination of opening, closing, and dilation.

e. Finally, using this algorithms background model is generated which will be used to compare with each one of the frames of the video.

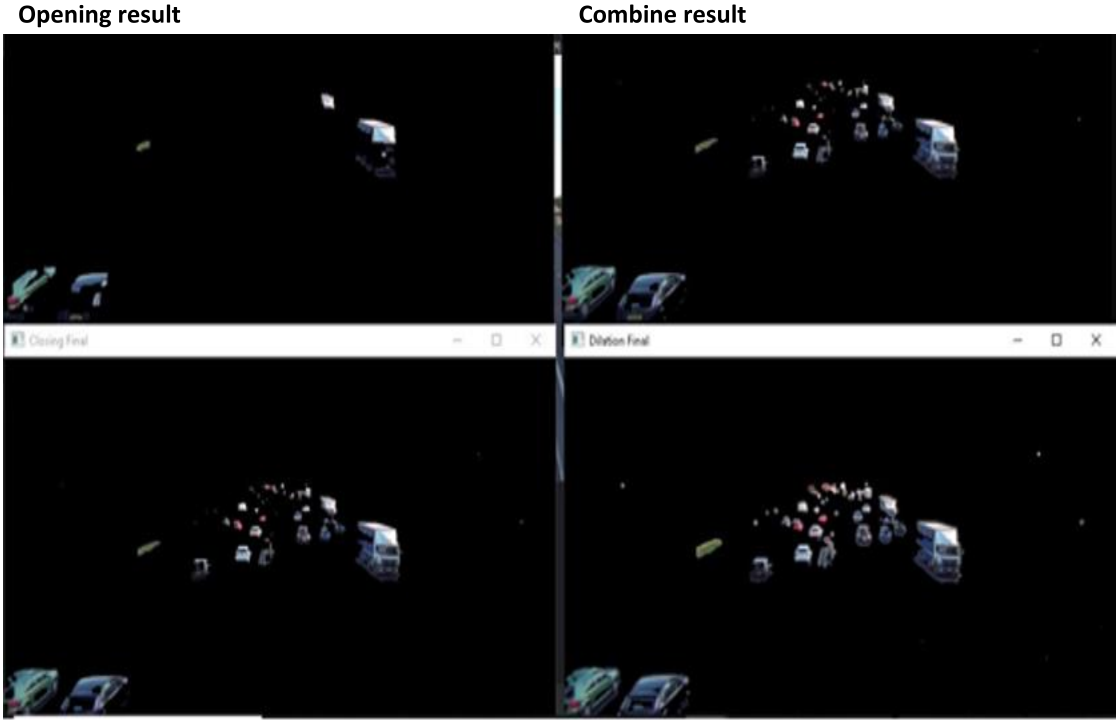


Figure 7: Morphological operations

The goal is to run all the algorithms and count how many white pixels are appearing in each of the frames for each of the algorithms, so we can measure how much motion each algorithm is detecting.

For example, if there is one algorithm that detects more objects it means that the number of white pixels is higher if compared with the other algorithms. This is the way we are going to implement to compare the algorithms and how much motion each one is detecting and motion is given by the white pixels.

5. Analysis and Result:

5.1 Comparison with people walking on road video

A 14-second video is captured. Process all the video frames and call all the algorithms simultaneously to count the number of white pixels in each frame. After counting, the result gets stored in a csv file (report.csv) in the format {Frame,Pixel Count}.

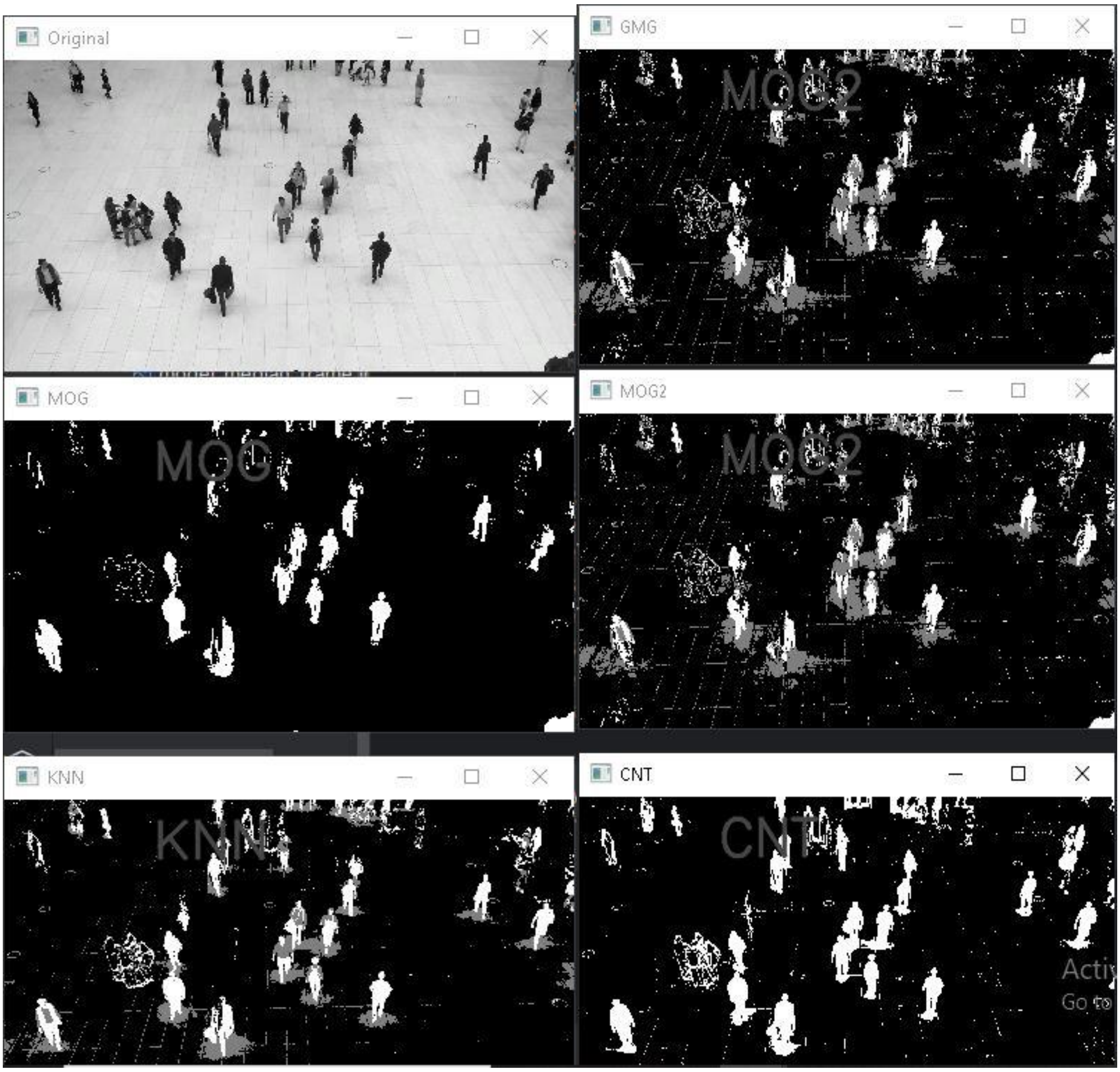
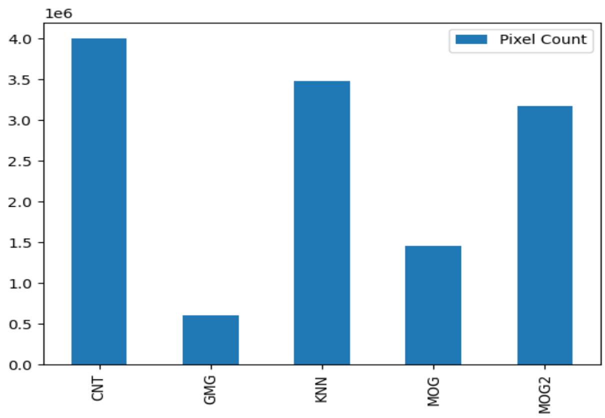


Figure 8: Background subtraction using the GMG, MOG2, MOG, KNN and CNT

After evaluating csv file, the result is, that CNT detects more white pixels, in second place KNN, in third place MOG2 then MOG, and finally GMG.



Frame	Pixel Count
CNT	3999571
GMG	601625
KNN	3477520
MOG	1458561
MOG2	3175124

Figure 9: Graph for number of white Pixel Detected in the Algorithm

Time taken by each algorithm to process the frames

The performance of the algorithm is the time taken by each algorithm to process some frames using the cv2.getTickCount() method. It is found that CNT is faster.

GMG	30.3749787
MOG	18.820174
MOG2	10.3576065
KNN	11.9015275
CNT	6.9811022

5.2 Comparison with the number of vehicles passing the road video

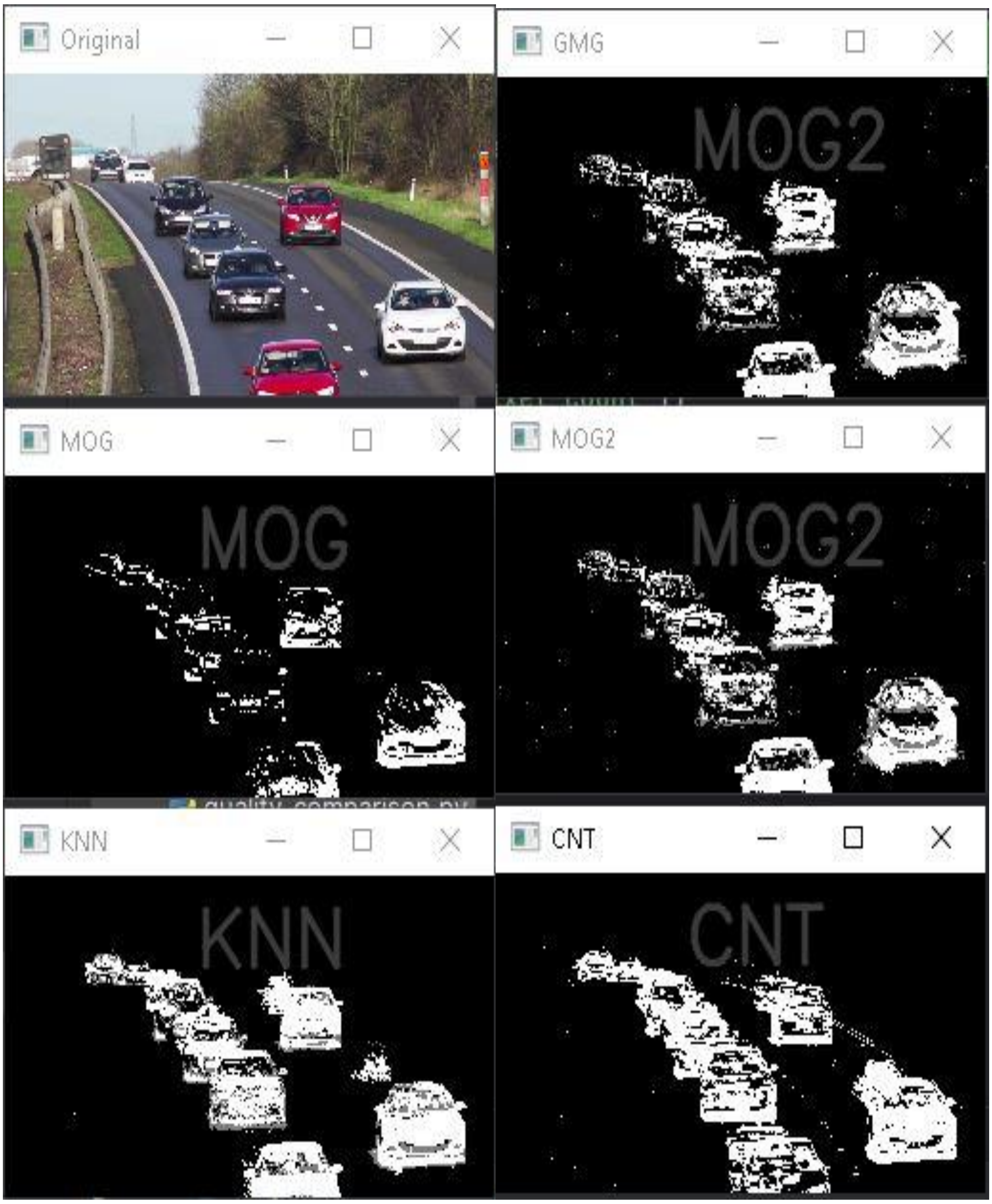
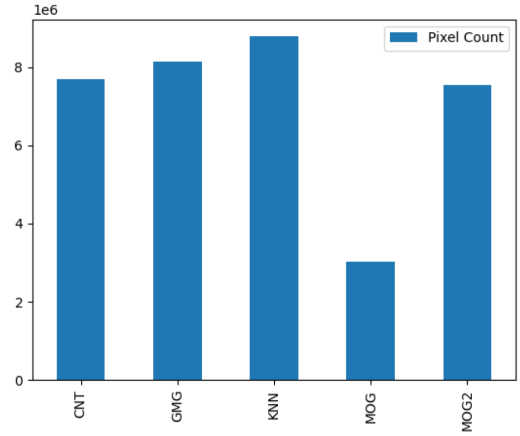
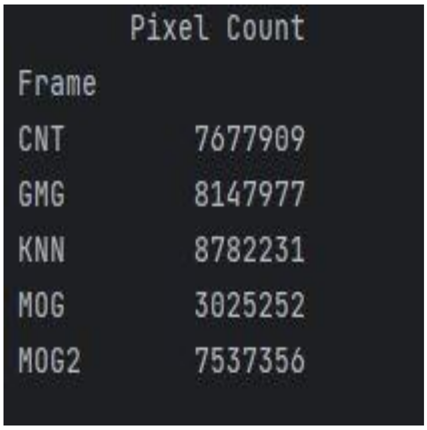


Figure 10: Background subtraction using the GMG, MOG2, MOG, KNN and CNT

After evaluating this video, the result is, that KNN detects more white pixels, in second place GMG, in third place CNT then MOG2, and finally MOG.





Pixel Count	
Frame	
CNT	7677909
GMG	8147977
KNN	8782231
MOG	3025252
MOG2	7537356

Figure 11: Graph for number of white Pixel Detected in the Algorithm

Time taken by each algorithm takes to process the frames

The performance of the algorithm shows CNT is faster.

GMG	14.508662
MOG	8.8681653
MOG2	5.3406265
KNN	6.7929019
CNT	3.6816676

6. Conclusion and Future Enhancement

In the output, Black pixels are the background, and white pixels are the foreground of the moving object. The goal is to count white pixels in each frame detected by the algorithms, that is, how much motion is detected by each algorithm. If any of the algorithms detects more objects, the number of white pixels is higher than other algorithms. In the case of video1, CNN detects more, and GMG detects fewer white pixels, whereas, in the second input, KNN detects more, and MOG2 detects fewer white pixels. The comparison of the performance, that is, each algorithm takes time to process some frames, concluded that CNN is faster among all the algorithms for both inputs.

Time-wise analysis is the same in both cases and shows the following sequence

CNT < MOG2 < KNN < MOG < GMG

In this research paper, static images are used to compare the algorithms. In future enhancement, this technique can be applied when the background is dynamic.

References

[1] A. N. Handayani, D. F. Laistulloh, I. A. E. Zaeni, M. Efendi, R. A. Asmara ,O. Fukuda. “Motion Detection for Children With Cerebral Palsy Using K-Nearest Neighbor.” *International Conference on Electrical and Information Technology (IEIT), Malang, Indonesia, 2022* (2022): 65-68.

[2] C. Sánchez-Ferreira, J. Y. Mori , C. H. Llanos. “Background subtraction algorithm for moving object detection in FPGA.” *VIII Southern Conference on Programmable Logic, Bento Gonçalves, Brazil, 2012* (2012): 1-6.

[3] Hanchinamani, Sunanda R., Sayantam Sarkar and Satish S Bhairannawar. “Design and Implementation of High Speed Background Subtraction.” *6th International Conference On Advances In Computing & Communications, ICACC 2016* (2016).

[4] Helly M Desai, Vaibhav Gandhi. “A Survey: Background Subtraction Techniques.” *International Journal of Scientific &*

Engineering Research 5.12 (December-2014): 1365.

[5] Jeisung Lee, Mignon Park. "An Adaptive Background Subtraction Method Based on Kernel Density Estimation." *Sensors* (2012).

[6] Jonathan Samuel Lumentut, Fergyanto E. Gunawan , Diana. "Evaluation of Recursive Background Subtraction Algorithms for Real-Time Passenger Counting at Bus Rapid Transit System." *International Conference on Computer Science and Computational Intelligence (ICCSCI 2015)* (2015): 445 – 453.

[7] Li Yao, Miaogen Ling. "An Improved Mixture-of-Gaussians Background Model with Frame Difference and Blob Tracking in Video Stream." *The Scientific World Journal* (2014).

[8] Limbasiya, Kalpesh and Pratik Ratanpara. "A Comprehensive Study on Motion Detection in Video with Surveillance System." *International Journal of Engineering Research & Technology (IJERT)* (April - 2014).

[9] Liu, Wei, et al. "Scene background estimation based on temporal median filter with Gaussian filtering." *23rd International Conference on Pattern Recognition (ICPR), Cancun, 2016* (2016): 132-136.

[10] Moch. Arief Soeleman, et al. "An Improvement for Background Modelling using a Mixture of Gaussian and Region Growing in Moving Objects Detection." *Journal of Physics* (2020).

[11] P. Wang, Y. Zhang , W. Jiang. "Application of K-Nearest Neighbor (KNN) Algorithm for Human Action Recognition." *IEEE 4th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), Chongqing, China, 2021* (2021): 492-496.

[12] Sanjeevkumar Angadi, Suvarna Nandyal. "A REVIEW ON OBJECT DETECTION AND TRACKING IN VIDEO SURVEILLANCE." *International Journal of Advanced Research in Engineering and Technology (IJARET)* (September 2020).

[13] Trnovszký, Tibor, Peter Sýkora and Róbert Hudec. "Comparison of background subtraction methods on near Infra-Red ." *Procedia Engineering* (2017): 192.

[14] V., Rekha and Natarajan K. "Foreground algorithms for detection and extraction of an object in multimedia." *International Journal of Electrical and Computer Engineering* (2020): 1849-1858.

[15] Wang, P., Y. Zhang and W. Jiang. "Application of K-Nearest Neighbor (KNN) Algorithm for Human Action Recognition." *IEEE 4th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), Chongqing, China, 2021* (2021): 492-496.

[16] Wei Liu, Yuanzheng Cai, Miaohui Zhang, Hui Li , Hejin Gu. "Scene background estimation based on temporal median filter with Gaussian filtering." *23rd International Conference on Pattern Recognition (ICPR), Cancun, 2016* (2016): 132-136.

[17] Zheng Yi, Fan Liangzhong. "Moving object detection based on running average background and temporal difference." *IEEE International Conference on Intelligent Systems and Knowledge Engineering, Hangzhou, China, 2010* (2010): 270-272.

[18] https://opencv.org