International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

A Hybrid Approach for Background Subtraction in Video: Combining RPCA, LBP, and Grassmann Average

Jeevith S H[1], Lakshmikanth S[2], Sandeep Kumar K[3]

Submitted: 28/03/2023 **Revised:** 27/05/2023 **Accepted:** 10/06/2023

Abstract: Background subtraction from moving video faces problems such as the complexity of the background, its movement and the change in light intensity arise and fragmented object make it difficult to detect moving objects in video. This paper presents a Novel hybrid model using Robust Principal Component Analysis (RPCA) and LBP (Local Binary Pattern) for background subtraction using Grassmann Average. Grassmann Average (GA) reduces the big outliers and RPCA gives sparse matrix (foreground information) and low rank matrix (back ground information). Feature extraction is done by using Local Binary Pattern (LBP). Finally, proposed RPCA-GA algorithm is executed in CD Net dataset. The results of proposed method are compared with various methods and also yields high Precision and Recall.

Keywords *– Histogram equalization, Local Binary Pattern, Robust Principal Component Analysis, Grassmann Average, etc.*

1. Introduction

Background subtraction plays a vital role in detecting moving objects, enabling automated video analytics in applications such as traffic monitoring, driving, and fire detection. The technique involves creating a background model without any moving objects and comparing it with each incoming video frame. Pixels that surpass a predefined threshold are identified as moving objects. This process serves as the foundation for accurate object detection and tracking.

Robust Principal Component Analysis (RPCA) [1] is a valuable approach for effectively reducing the static background in video sequences. By reconstructing low-rank matrices, RPCA is capable of simulating the static background component, while the remaining component represents the moving objects. However, it is important to note that the low-rank output of RPCA closely resembles the low-level matrix, limiting its ability to capture dynamic background variations. Additionally, in datasets with substantial outliers, the performance of RPCA can be affected due to the large presence of these outliers.

Principal Component Analysis (PCA) is widely recognized as one of the most prevalent techniques for dimensionality reduction, modeling, and data analysis across various scientific fields and applications. It serves as a fundamental tool in computer vision. While there are methods available to adapt PCA to large datasets, a persistent challenge remains unresolved. Large datasets are often generated automatically and are too vast for manual validation, leading to the inclusion of significant outliers within the so-called "big data" paradigm. While there exist solutions that enhance the robustness of PCA against outliers [2-5], they are generally not well-suited for large datasets. In general, methods that prioritize robustness in PCA lack scalability, while scalable PCA methods tend to sacrifice robustness. In this research, the authors introduce a novel formulation and scalable algorithm for robust PCA that surpasses the performance of previous methods. The primary contribution lies in formulating the subspace estimate as the mean computation of the subspace.

The research work delves into the development of the mean operator on the Grassmann manifold [6], which formalizes the concept of the mean subspace encompassed by the data. This approach, known as Grassmann Average (GA), exhibits a connection to standard PCA and demonstrates that, for Gaussian data, the subspace derived through GA aligns with the subspace obtained via standard PCA. In the context of background subtraction, when moving objects are detected, a low-rank matrix derived from the RPCA process undergoes processing to model the background. By leveraging GA, the mean subspace estimation allows for more accurate and reliable background modeling, further enhancing the effectiveness of the overall RPCA-GA algorithm in capturing foreground objects and isolating the stationary background.

[1] *Assistant Professor, Dept. of E&TE,SSIT , Tumakuru,India*
[2] *Associate Professor, Dept. of. ECE, Acharya Institute of Technology, Bengaluru, India*
[3] *Assistant Professor, Dept. of ECE, Acharya Institute of Technology, Bengaluru, India*

The study begins by providing a comprehensive literature review in Section II, highlighting the relevant research in the field. In Section III, the proposed methods are explained in detail. The results of the MATLAB-based simulation are presented and analyzed in Section IV. Finally, the paper concludes with a summary of the findings and conclusions in Section V.

2. Literature Review

The motion detection literature is presented below and shows some of the important methods for detecting an area of a CCTV scene.

However, motion detection suffers from a number of problems caused by noise sources, complex backgrounds, variations in scene lighting, and shadows from stationary and moving objects. For this reason, several methods have been proposed to solve this problem by keeping only the moving objects of interest. These methods fall into three categories: background reduction, time difference, and optical flux [7-18].

The time difference is highly adaptable and dynamic depending on the environment; however, it does a poor job of fetching all the relevant functional pixels. Therefore, to obtain the desired shape of a moving object, methods based on morphological operations and space filling are used. On the other hand, background reduction provides the most complete asset data, but is sensitive to dynamic changes due to lighting and extraneous events. Background reduction techniques have been classified as follows: basic background modeling, statistical background modeling, fuzzy background modeling, background clustering, neural network background modeling, background wave modeling, and background estimation [19-24]. In turn, the optical flow method can also be used to detect moving objects. However, most optical streaming techniques are computationally complex and cannot be applied to full-screen real-time video streams without dedicated hardware [25].

In [26], a method is shown that increases the frame differences by first sorting the blocks in the background and then using the correlation coefficients. They used a pixel-level block classification technique to detect the movement of people in various scenes from the dataset (Wall eur and I2R) with an average accuracy of 0.6396%.

In turn, in [27] they present a robust approach to background triggering based on superpixel motion detection. The spatial and temporal properties of the frame are adjusted to remove foreground objects. First, select a partial sequence with stable lighting conditions to get closer to the background, then segment the image into superpixels to preserve texture and filter out foreground objects. They used the SBMnet dataset, which contained 8 categories, which, unlike other methods in the prior art, achieved the best average F-score in the low frame rate category with a score of 0.7222.

In [28], they demonstrated an algorithm based on W4 and frame difference that overcomes the inadequacy of false detection due to background mutations; as well as elimination of gaps caused by differences in frames. This algorithm is used to detect security issues such as illegal intrusions, persistent warnings, and illegal movements. Three infrared categories (TM, SC and ROD) of DM642 were analyzed, with results of 0.99 for the DIN sequence and 0.9608 for the ROA scene.

In [29], they proposed a method that searches for areas of dynamic background by analyzing video from CCTV cameras and helps to eliminate false alarms. This was estimated using the 2012/2014 CDnet dataset with an average accuracy of 0.8650 for CDnet2012 and 0.7668 for CDnet2014.

In [30], they analyzed methods for reducing the background, frame difference and SOBS; for detecting moving objects from CCTV cameras. In the end, they did not provide any numerical results, but did mention that the best way to find motion is to use background subtraction, because the screen difference method has the disadvantage of detecting objects with uniform intensity distribution values. On the other hand, the SOBS method gives good results, but the processing time is very long.

In a study [31] based on motion detection, which presents 12 methods based on motion detection, which demonstrated different methods of motion detection, such as: time differentiation (frame difference), three-plane difference (3FD), adaptive background (middle filter), forgetting morphology temporal gradient (FMTG), background estimation, space-time Markov field, Gaussian mean (RGA), Gaussian mixing (MoG), space-time image entropy (STEI), space-time image entropy difference (DSTEI) , clear background (Eig-Bg) and simplified self-organizing map (Simp-SOM). In the end, the best results were analyzed using the CDnet2014 dataset, it was the result classified by the GMM with a specificity of 0.99993, 3.08499 in PWC and an accuracy of 0.61021. The lowest efficiency of the motion detection method is STEI with a specificity of 0.78646, PWC 22.18321 and an accuracy of 0.12881.

3. Proposed Methodology

The experiment computes a background model with different video to observe the effectiveness of different RPCA approaches and the effect of preprocessing on the results. Figure 1 shows histogram equalization as a preliminary process for each raw frame obtained from video. Various frames are extracted from video and then processed through LBP features to get texture features. RPCA is applied to differentiate the foreground and background as matrix decomposition. Further RPCA projection is scaled by Grassmann Average to get low rank matrix contains the background and sparse matrix contains the foreground information.

The foreground was obtained by post-processing as given in Figure 1 the sparse matrix. Post-processing is the same for all attempts. These measures include hard boundaries, morphological filters, and active boundaries with negative compression shears [32].

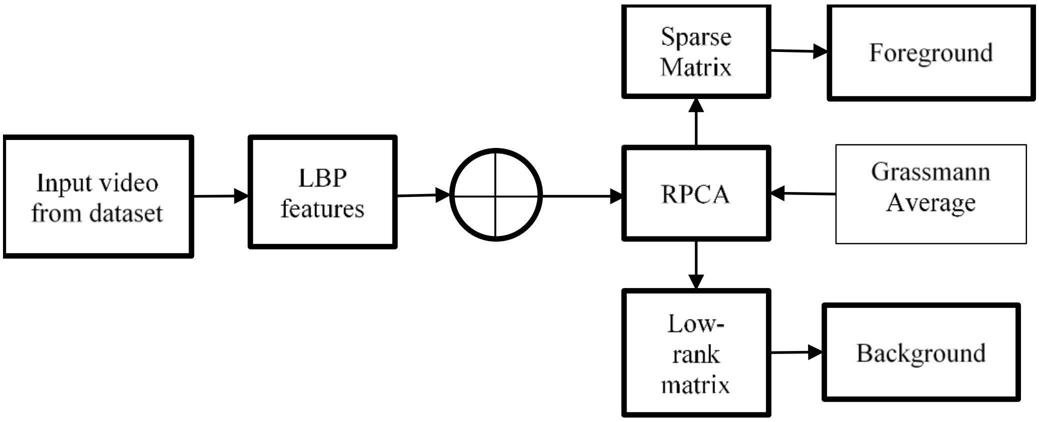


Fig 1: Block diagram for pre-processing phase

The novel hybrid model combines Robust Principal Component Analysis (RPCA) and Local Binary Pattern (LBP) techniques for background subtraction using Grassmann Average (GA). The goal is to accurately separate the foreground objects from the background in a given video sequence. The Grassmann Average (GA) is employed to reduce the impact of big outliers, which often result from sudden changes in lighting conditions or moving objects that do not belong to the foreground. This step helps in improving the robustness of the background subtraction process. RPCA is utilized to decompose the input video frames into a sparse matrix representing the foreground information and a low-rank matrix representing the background information. The sparse matrix captures the salient moving objects in the scene, while the low-rank matrix models the stationary background. To perform feature extraction, Local Binary Pattern (LBP) is applied to each frame. LBP is a texture descriptor that encodes the local texture patterns of an image. By extracting LBP features, the model can capture and represent the textural characteristics of the foreground and background regions. The proposed RPCA-GA algorithm is then executed on the CD Net dataset, which is a benchmark dataset widely used for evaluating background subtraction algorithms. The dataset contains various challenging scenarios, such as dynamic backgrounds, camera jitter, and intermittent object motion. Rest of methodology is explained in following subheadings.

A. Feature Extraction using Local Binary Pattern (LBP)

Feature extraction plays a crucial role in background subtraction from moving video frames. In this case, Local Binary Pattern (LBP) is utilized as a texture descriptor to capture the local texture patterns of the foreground and background regions. LBP computes a binary code for each pixel by comparing its intensity value with its neighboring pixels. The extracted LBP features are then used to represent the textural characteristics and discriminate between the foreground and background regions.

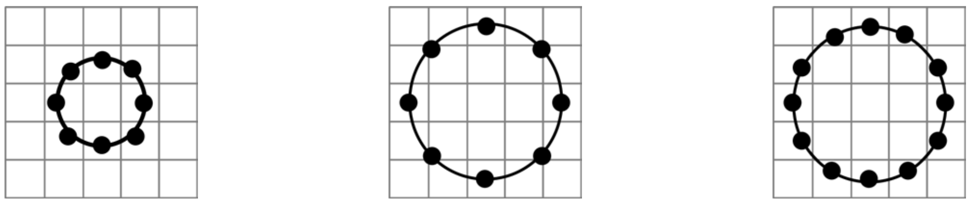
LBP is a pattern recognition method with high binary coding-based discrimination between each pixel in the image and its neighbors. The LBP operator transforms the pixels of the given image into a sequence of 1 or 0 values, as shown in Figure 2, by matching the value of each pixel in the 3×3 neighborhood of a selected pixel in the image with the selected pixel. A label is created for each pixel of the image with a label where the generated binary number: 11001010 is converted to 83.

$$\begin{bmatrix} 31 & 129 & 214 \\ 74 & 40 & 15 \\ 9 & 60 & 38 \end{bmatrix} \quad \overset{\textbf{Thresholding}}{\begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}} \times \overset{\textbf{Coding}}{\begin{bmatrix} 2^7 & 2^0 & 2^1 \\ 2^6 & 0 & 2^2 \\ 2^5 & 2^4 & 2^3 \end{bmatrix}} = \begin{bmatrix} ? & ? & ? \\ ? & 83 & ? \\ ? & ? & ? \end{bmatrix}$$

$$2^0 + 2^1 + 2^4 + 2^2 = 83$$

Fig 2: Implementation of the LBP operator

The main disadvantage of the basic local binary operator is that it estimates a small space around a 3×3 matrix. Thus, for large structures, significant features can be neglected. The extended LBP operator (P, R) is represented by the area of a circle, where P is the number of pixels in the vicinity of the circle and R is the radius of the area of the circle. Thus, it was possible to perform texture analysis of images of different sizes more effectively. An example of different LPB operators is given in Figure 3.



(a) R = 1, P = 8 (b) R = 2, P = 8 (c) R = 2, P = 12

Fig 3: Neighbor set's circular symmetry

Mathematical Formulation:

LBP is computed for each pixel in an image by comparing its intensity value with its neighboring pixels. The LBP value for a pixel is calculated as follows:

$$LBP(x_c) = \sum_{i=0}^{P-1} s(I(x_i) - I(x_c))2^i \quad (1)$$

Here,

x_c represents the central pixel.

x_i denotes the neighboring pixels.

P is the number of neighboring pixels.

$s(.)$ is the sign function (1 if the argument is non-negative, otherwise 0).

$I(.)$ denotes the intensity value of a pixel.

Pseudo Code for Feature Extraction using LBP:

1. $Input: Video\ frames\ sequence\ F = \{F_1, F_2, \ldots, F_T\}\ (T\ frames)$

2. $Output: LBP\ features\ sequence\ LBP = \{LBP_1, LBP_2, \ldots, LBP_T\}\ (T\ frames)$

3. $Function\ LBP_{Extraction}(image)$:

   3.1. $Convert\ the\ image\ to\ grayscale.$

   3.2. $Initialize\ an\ empty\ LBP\ feature\ vector\ LBP_{vec}.$

   3.3. $For\ each\ pixel\ (x, y)\ in\ the\ image$:

      3.3.1. $Retrieve\ the\ intensity\ value\ of\ the\ central\ pixel\ I(x, y).$

      3.3.2. $Initialize\ a\ binary\ code\ LBP_{code}\ to\ 0.$

      3.3.3. $For\ each\ neighboring\ pixel\ (x_i, y_i)\ around\ (x, y)$:

         3.3.3.1. $Retrieve\ the\ intensity\ value\ of\ the\ neighboring\ pixel\ I(x_i, y_i).$

3.3.3.2. $Update\ the\ binary\ code\ LBP\_code\ based\ on\ the\ comparison\ of\ intensities$:

$$LBP_{code} = LBP_{code}\ OR\ \left(s\left(I(x_i, y_i)\ -\ I(x, y)\right) << i\right) \tag{2}$$

3.3.4. $Append\ the\ LBP_{code}\ to\ the\ LBP_{vec}$.

3.4. $Return\ the\ LBP_{vec}$.

4. $Procedure\ FeatureExtraction(F)$:

4.1. $Initialize\ an\ empty\ LBP\ features\ sequence\ LBP$.

4.2. $For\ each\ frame\ F_t\ in\ F$:

4.2.1. $Apply\ the\ LBP_{Extraction}\ function\ on\ the\ frame\ F_t\ to\ obtain\ the\ LBP\ features\ LBP_t$.

4.2.2. $Append\ LBP_t\ to\ the\ LBP\ sequence$.

4.3. $Return\ the\ LBP\ sequence$.

5. $Call\ FeatureExtraction(F)\ to\ extract\ LBP\ features\ from\ the\ video\ frames\ sequence\ F$.

The above pseudo code outlines the process of feature extraction using LBP for background subtraction from moving video frames. It iterates over each pixel in each frame, computes the LBP code by comparing intensities with neighboring pixels, and constructs a sequence of LBP features for each frame. These extracted features can then be used for subsequent tasks such as classification, object recognition, or anomaly detection.

B. Robust Principal Component Analysis

Robust Principal Component Analysis (RPCA) is a powerful technique used for background subtraction from moving video frames. RPCA aims to decompose the input video into a low-rank matrix representing the background information and a sparse matrix capturing the foreground objects or anomalies. By separating the background and foreground components, RPCA enables accurate detection and segmentation of moving objects, such as faces, in the video.

Mathematical Formulations:

Robust Principal Component Analysis (RPCA): It is a technique used to decompose a given matrix into a low-rank component and a sparse component. In the context of background subtraction from moving video frames, RPCA is employed to separate the background information (low-rank component) from the foreground objects or anomalies (sparse component). The underlying assumption is that the majority of the video frames belong to the background, which can be modeled as a low-rank matrix, while the foreground objects or anomalies introduce sparsity in the matrix representation.

Mathematically, given an input video matrix $X \in R_d \times T$, where d represents the number of pixels and T represents the number of frames, RPCA aims to find the decomposition:

$$\min(\|L\|^* + \lambda^*\|S\|_1) \quad subject\ to\ X = L + S \tag{3}$$

Where, L denotes the low-rank matrix representing the background, S represents the sparse matrix capturing the foreground objects or anomalies, $\|L\|^*$ is the nuclear norm of matrix L (a convex relaxation of the rank function), $\|S\|_1$ is the $\ell 1$ norm of matrix S, and λ is a regularization parameter that controls the trade-off between the low-rank and sparse components. By solving this optimization problem, RPCA separates the background and foreground components, enabling accurate background subtraction.

Singular Value Thresholding (SVT): It is a key step in the RPCA algorithm, specifically in enforcing the low-rankness of the background matrix. The idea behind SVT is to perform soft-thresholding on the singular values of the matrix, effectively shrinking small singular values towards zero. By applying SVT, the low-rank component is enhanced while promoting sparsity in the sparse component.

Mathematically, given an input matrix L and a threshold value τ, the singular value thresholding operation $SVT(L, \tau)$ is performed as follows:

$$SVT(L, \tau) = U * diag(\sigma_{threshold}) * V^T \tag{4}$$

Where, U and V are the left and right singular vectors of L, and $\sigma_{threshold}$ represents the singular values after soft-thresholding. By setting small singular values below the threshold τ to zero and keeping the larger singular values, SVT helps to emphasize the low-rank structure of the background matrix.

Sparse Component Recovery: It is the process of reconstructing the sparse matrix S, which captures the foreground objects or anomalies, given the observed video frames and the estimated low-rank background matrix L. The goal is to recover the sparse component by solving an $\ell 1$-minimization problem, which promotes sparsity in the foreground representation.

Mathematically, the sparse component recovery problem is formulated as follows:

$$\min \|S\|_1, \text{subject to } X = L + S \quad (5)$$

Where, X represents the observed video frames and L is the estimated low-rank background matrix. The optimization problem seeks to find the sparse matrix S that minimizes the $\ell 1$ norm while satisfying the constraint that the sum of the background matrix L and the sparse matrix S equals the observed frames X. By solving this optimization problem, the sparse component is effectively recovered, allowing for accurate foreground object detection and segmentation.

Pseudo Code for Robust Principal Component Analysis (RPCA):

1. $Input: Video\ frames\ sequence\ F = \{F_1, F_2, \ldots, F_T\}\ (T\ frames)$

2. $Output: Low - rank\ background\ matrix\ L, Sparse\ foreground\ matrix\ S$

3. $Procedure\ RPCA(F, \lambda, \tau)$:

3.1. $Initialize\ matrices\ L\ and\ S\ with\ zeros\ of\ the\ same\ dimensions\ as\ the\ input\ frames.$

3.2. $Set\ the\ maximum\ number\ of\ iterations\ N\ and\ convergence\ threshold\ \varepsilon.$

3.3. $For\ iteration\ k = 1\ to\ N$:

3.3.1. $Compute\ the\ singular\ value\ decomposition\ (SVD) of\ matrix\ X = F - S + L$:

$$X = U * \Sigma * V^T$$

3.3.2. $Apply\ singular\ value\ thresholding\ (SVT)\ to\ the\ singular\ values$:

$$\Sigma_{threshold} = Soft\ Threshold(\Sigma, \tau)$$

3.3.3. $Update\ the\ background\ matrix\ L$:

$$L = U * \Sigma_{threshold} * V^T$$

3.3.4. $Update\ the\ sparse\ matrix\ S$:

$$S = Soft\ Threshold(F - L, \lambda)$$

3.3.5. $Check\ the\ convergence\ criterion$:

$$If\ \|X - L - S\|_F \leq \varepsilon, break\ the\ iteration\ loop.$$

3.4. $Return\ the\ background\ matrix\ L\ and\ the\ sparse\ matrix\ S.$

4. $Call\ RPCA(F, \lambda, \tau) to\ perform\ Robust\ Principal\ Component\ Analysis\ on\ the\ video\ frames$

$sequence\ F, with\ regularization\ parameter\ \lambda\ and\ singular\ value\ threshold\ \tau.$

The above pseudo code outlines the process of Robust Principal Component Analysis (RPCA) for background subtraction from moving video frames. It iteratively updates the low-rank background matrix L and the sparse foreground matrix S by applying singular value thresholding and $\ell 1$-minimization. This decomposition enables the separation of the background and foreground components, facilitating accurate background subtraction and foreground object detection.

C. Robust Principal Component Analysis with Grassmann Average

The RPCA-GA (Robust Principal Component Analysis with Grassmann Average) algorithm combines Robust Principal Component Analysis (RPCA) with the Grassmann Average (GA) technique to perform background subtraction in video sequences. This algorithm aims to separate the background information (low-rank component) from the foreground objects (sparse component) while incorporating the Grassmann Average for enhanced robustness. The following is a detailed description, including mathematical formulations and pseudo code, for the RPCA-GA algorithm.

1. Problem Formulation

Given an input video matrix $X \in R_d \times T$, where d represents the number of pixels and T represents the number of frames, the goal is to decompose X into a low-rank matrix L (representing the background) and a sparse matrix S (representing the foreground) using RPCA. The Grassmann Average (GA) technique is then incorporated to improve the robustness of the background subtraction.

2. *RPCA-GA Algorithm*

2.1. Initialization

- Set the maximum number of iterations N and convergence threshold ε.
- Initialize the low-rank matrix L and sparse matrix S with zeros.
- Initialize the background model B_{avg} with an arbitrary frame from the video.

2.2. RPCA Decomposition

- For iteration $k = 1$ to N:
  - Update the sparse matrix S by solving the $\ell 1$-minimization problem:

$$S = \arg\min \|S\|_1 \ , subject\ to\ X = L + S \tag{6}$$

  - Update the low-rank matrix L by applying singular value thresholding (SVT):

$$L = SVT(X - S, \tau) \tag{7}$$

    Where, SVT represents the singular value thresholding operation.
  - Check the convergence criterion: If the change in L and S is smaller than ε, break the iteration loop.

2.3. Grassmann Average Update

- Compute the Grassmann Average of the background models:

$$B_{avg} = Grassmann\ Average(\{B_1, B_2, \dots, B_k\}) \tag{8}$$

  Where, $\{B_1, B_2, \dots, B_k\}$ represents the set of background models obtained during the iterations.

2.4. Foreground Detection

- Compute the residual matrix $R = X - B_{avg}$.
- Apply a thresholding operation to identify the foreground pixels:

$$F\ =\ Threshold(R) \tag{9}$$

  Where, F represents the binary foreground mask.

3. *Pseudo Code for RPCA-GA Algorithm:*

Input: Video matrix $X \in R_d \times T$, *convergence threshold* ε, *maximum iterations* N, *threshold* τ.

Output: Binary foreground mask F.

Procedure $RPCA - GA(X, \varepsilon, N, \tau)$*:*

Initialization:

Initialize L, S, B_{avg}, *and set* $k\ =\ 1$.

RPCA Decomposition:

For iteration $k = 1$ *to* N*:*

Update S*:*

$S = \arg\min \|S\|_1 \ , subject\ to\ X = L + S.$

Update L*:*

$L = SVT(X - S, \tau)$, *where SVT represents singular value thresholding.*

Check convergence:

If $\|L_{old} - L\| \leq \varepsilon$ *and* $\|S_{old} - S\| \leq \varepsilon$, *break the loop.*

$Update\ L_{old} = L\ and\ S_{old} = S.$

Grassmann Average Update:

$B_{avg} = Grassmann\ Average(\{B_1, B_2, \dots, B_k\})$, *where* $\{B_1, B_2, \dots, B_k\}$ *represents the set of background models.*

Foreground Detection:

R = X - B_avg.

F= Threshold(R).

Return F.

The RPCA-GA algorithm begins by initializing the necessary variables and matrices. The low-rank matrix L and sparse matrix S are set to zero, and the background model B_{avg} is initialized with an arbitrary frame from the video.

The algorithm then proceeds with the RPCA decomposition. It iterates from $k = 1$ to N, where N is the maximum number of iterations. In each iteration, the sparse matrix S is updated by solving the $\ell 1$-minimization problem, aiming to minimize the $\ell 1$-norm of S while satisfying the constraint $X = L + S$. The low-rank matrix L is then updated using the singular value thresholding (SVT) operation, where the difference between X and S is thresholded based on the parameter τ.

After each iteration, the convergence criterion is checked. If the change in L and S is smaller than the convergence threshold ε, the iteration loop is broken.

Next, the Grassmann Average update step is performed. The background models obtained during the iterations, represented as $\{B_1, B_2, \dots, B_k\}$, are used to compute the Grassmann Average B_{avg}. This step enhances the robustness of the background subtraction process.

Finally, the foreground detection is carried out. The residual matrix R is computed as the difference between the input video matrix X and the background model B_{avg}. A thresholding operation is applied to R to identify the foreground pixels, resulting in the binary foreground mask F.

The input parameters are the video matrix X, convergence threshold ε, maximum iterations N, and threshold τ. The algorithm returns the binary foreground mask F.

The RPCA-GA algorithm combines the strengths of RPCA and GA to improve background subtraction in video sequences. The integration of RPCA separates the video frames into low-rank and sparse components, while the Grassmann Average reduces the impact of outliers and provides a robust estimate of the background. This hybrid approach enhances the accuracy and robustness of foreground object detection and segmentation, making it suitable for various computer vision applications such as surveillance, video analytics, and object tracking.

4. Simulation Results

A comparative analysis of four background subtraction techniques using the CD Net dataset [31] is presented. The database contains 1700 frames with a resolution of 320×240, the first 100 frames are used for background initialization, and the rest of the images are used for background updates for object detection. Each frame has a separate ground truth. Four techniques of background subtraction are compared. Following are the evaluation parameters:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{12}$$

$$Specificity = \frac{TN}{TN + FN} \tag{13}$$

$$Error\ Rate = \frac{FP + FN}{TP + TN + FP + FN} \tag{14}$$

$$False\ Positive\ Rate\ (FPR) = \frac{FP}{FP + TN} \tag{15}$$

$$F - Score = \frac{2TP}{2TP + FP + FN} \tag{16}$$

$$Matthews\ Correlation\ Coefficient\ (MCC) = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FN)(TP + FP)(TN + FN)(TN + FP)}} \tag{17}$$

$$Kappa\ Statistics = \frac{Observed\ accuracy - expected\ accuracy}{1 - expected\ accuracy} \tag{18}$$

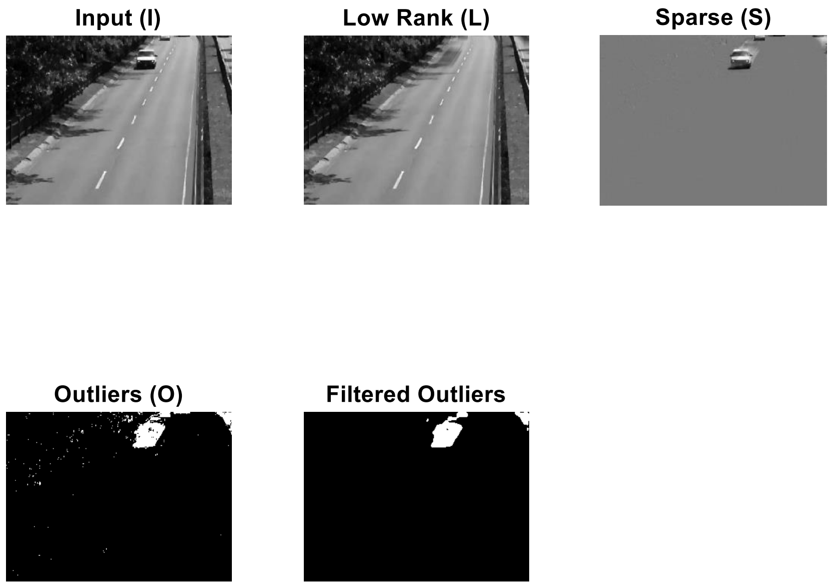


Fig 4: Simulation performed on Highway image-1

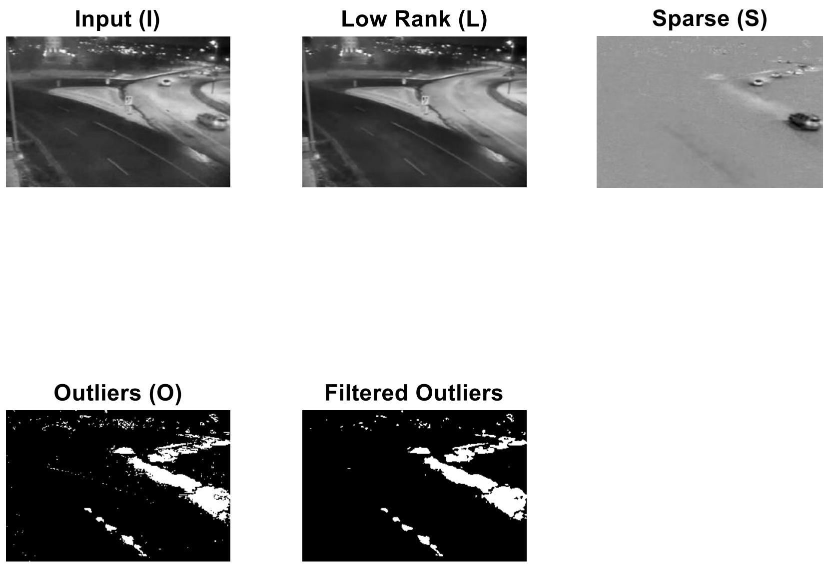


Fig 5: Simulation performed on Highway image-2

The proposed method will be evaluated against another benchmark method. The reference videos “Overpass”, “Fountain 02”, “Canoe” and “Boats” were taken from the CDNet dataset [31].

Table 1: Information of benchmark videos from CDNet dataset

	Overpass	Fountain02	Canoe	Boats
Frame size	240 × 320 × 3	288 × 432 × 3	240 × 320 × 3	240 × 320 × 3
Start-end	2001-3000	1-1000	101-1100	7000-7999
Dynamic texture	Waving tree	Fountain	Flowing water	Flowing water

"Start-end means the frame number of the start frame and the end frame of a full sequence"

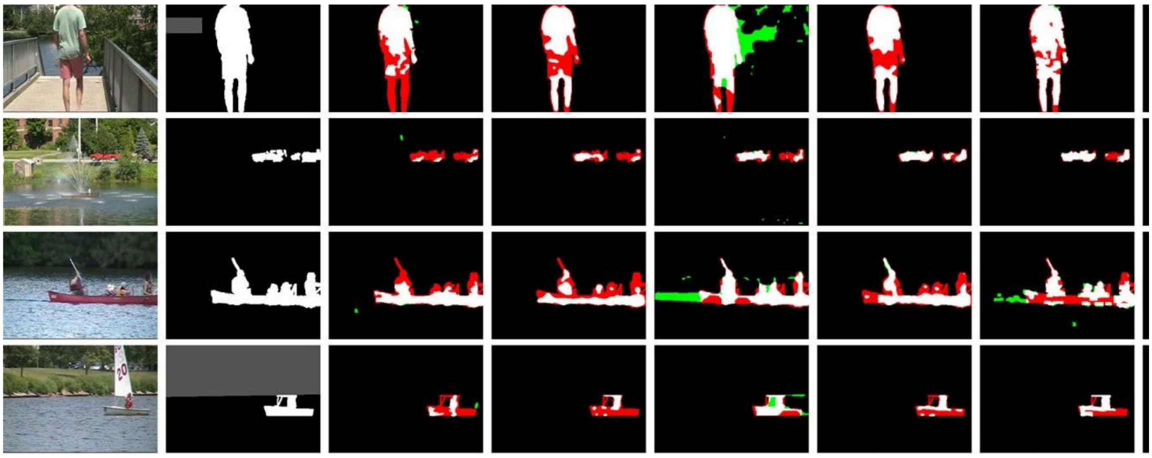


Fig 6: Comparative results with different images

Table 2: Comparative results

[%]	CANDID [33]	SuBSENSE [34]	CD Net [35]	Proposed method
		Overpass		
Precision	87.24	95.21	93.29	95.66
Recall	98.81	97.52	98.40	99.69
F-Score	92.67	96.35	95.78	97.63
		Fountain02		
Precision	71.80	89.65	85.65	78.53
Recall	27.92	27.81	69.70	63.35
F-Score	40.21	42.45	76.86	70.13
		Canoe		
Precision	49.04	81.29	57.15	89.91
Recall	54.99	57.46	63.07	90.56
F-Score	51.85	67.33	59.97	90.23
		Boats		
Precision	94.52	96.67	88.97	96.34
Recall	77.78	94.99	99.88	87.37
F-Score	85.34	95.83	94.11	91.63
		Average		
F-Score	67.51	75.49	81.68	87.41

For CD Net data, the authors of [35] generated several false positive (FP) pixels, but also several true positive (TP) pixels, which represent low F-score values. BF-LRGB showed the second best specificity in “Overpass, Canoe, and Boats”, but produced many false positive pixels in “Overpass”. Methods based on CANDID [33] and SuBSENSE [34] samples yielded more false positive pixels in all CD Net videos. For precision, they took first and second places in all video tests, except for “Boats”.

5. Conclusion

This article introduces a new dynamic background reduction technique known as RPCA-GA. The proposed method estimates the background by minimizing the multiplication of the term “background mask” and “rank”. The computational costs due to the iteration of the proposed method are almost linear. Although this method does not yield the best quantitative results in all respects compared to the previous method, it exceeds the average F-score of other algorithms for the dataset (CD Net dataset). Different methods are implemented SuBSENSE and CANDID methods due to balanced high precision (96.34% in proposed method) and recall (99.69% in proposed method). It can be seen F-Score of overpass is 97.63% as compared to other methods.

References

[1] Peng.C, Chen. Y, Kang.Z, Chen. C,Cheng, Q, Robust principal component analysis: A factorization-based approach with linear complexity. *Information Sciences 513*, 2020, pp.581-599.

[2] Vaswani, N, Bouwmans, T, Javed, S, Narayanamurthy, P, Robust subspace learning: Robust PCA, robust subspace tracking, and robust subspace recovery. *IEEE signal processing magazine*, *35*(4), 2018, pp.32-55.

[3] Vaswani, N, Narayanamurthy P, Static and dynamic robust PCA and matrix completion: A review. *Proceedings of the IEEE*, *106*(8), 2018, pp.1359-1379.

[4] Chen Y, Fan J, Ma C, Yan Y, Bridging convex and nonconvex optimization in robust PCA: Noise, outliers, and missing data. *arXiv preprint arXiv:2001.05484*, 2020.

[5] Yang J.H, Zhao X.L, Ji T.Y, Ma T.H, Huang, T.Z, Low-rank tensor train for tensor robust principal component analysis. *Applied Mathematics and Computation*, *367*, 2020, p.124783.

[6] Minnehan, B, Savakis, A, Grassmann Manifold Optimization for Fast $ L_1 $-Norm Principal Component Analysis. *IEEE Signal Processing Letters*, *26*(2), 2018, pp.242-246.

[7] Tripathi, R.K., Jalal A.S, Agrawal, S.C., Abandoned or removed object detection from visual surveillance: a review. *Multimedia Tools and Applications*, *78*(6), 2019, pp.7585-7620.

[8] Liu, K. and Ma, H., October. Exploring background-bias for anomaly detection in surveillance videos. In *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 1490-1499.

[9] Ansari, M. and Singh, D.K., Human detection techniques for real time surveillance: A comprehensive survey. *Multimedia Tools and Applications*, *80*(6), 2021, pp.8759-8808.

[10] Mabrouk, A.B. and Zagrouba, E, Abnormal behavior recognition for intelligent video surveillance systems: A review. *Expert Systems with Applications*, *91*,2018, pp.480-491.

[11] Goyal, K. and Singhai, J, Review of background subtraction methods using Gaussian mixture model for video surveillance systems. *Artificial Intelligence Review*, *50*(2), 2018,pp.241-259.

[12] Bouwmans, T., Javed, S., Sultana, M. and Jung, S.K., Deep neural network concepts for background subtraction: A systematic review and comparative evaluation. *Neural Networks*, *117*, 2019, pp.8-66.

[13] Azeez, B. and Alizadeh, F., February. Review and Classification of Trending Background Subtraction-Based Object Detection Techniques. In *2020 6th International Engineering Conference “Sustainable Technology and Development”(IEC)*, 2020, pp. 185-190). IEEE.

[14] Jenifa Sabeena, S. ., & Antelin Vijila, S. . (2023). Moulded RSA and DES (MRDES) Algorithm for Data Security. International Journal on Recent and Innovation Trends in Computing and Communication, 11(2), 154–162. https://doi.org/10.17762/ijritcc.v11i2.6140

[15] Zhang, T., Li, J., Jia, W., Sun, J. and Yang, H., Fast and robust occluded face detection in ATM surveillance. *Pattern Recognition Letters*, *107*, 2018,pp.33-40.

[16] Sun, S., Kuang, Z., Sheng, L., Ouyang, W. and Zhang, W, Optical flow guided feature: A fast and robust motion representation for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018,pp. 1390-1399.

[17] Zhou, D., Frémont, V., Quost, B., Dai, Y. and Li, H., Moving object detection and segmentation in urban environments from a moving platform. *Image and Vision Computing*, *68*, 2017,pp.76-87.

[18] Đokić, L., Jokić, A., Petrović, M., Slavković, N. and Miljković, Z., Application of metaheuristic optimization algorithms for image registration in mobile robot visual control. *Serbian Journal of Electrical Engineering*, *18*(2), 2021, pp.155-170.

[19] Kushwaha, A., Khare, A., Prakash, O. and Khare, M., Dense optical flow based background

subtraction technique for object segmentation in moving camera environment. *IET Image Processing*, *14*(14), 2020, pp.3393-3404.

[20] Mohanty, S.K. and Rup, S., An adaptive background modeling for foreground detection using spatio-temporal features. *Multimedia Tools and Applications*, *80*(1), 2021, pp.1311-1341.

[21] Qiu, S. and Li, X., Moving target extraction and background reconstruction algorithm. *Journal of Ambient Intelligence and Humanized Computing*, 2020, pp.1-9.

[22] Zhang, K., Tong, S., Shi, H., Yue, G. and Zhao, J., Moving object detection of assembly components based on improved background subtraction algorithm. In *IOP Conference Series: Materials Science and Engineering* (Vol. 1009, No. 1, p. 012063), 2021,IOP Publishing.

[23] Maddalena, L. and Petrosino, A., Self-organizing background subtraction using color and depth data. *Multimedia Tools and Applications*, *78*(9), 2019, pp.11927-11948.

[24] Lu, X., Xu, C., Wang, L. and Teng, L., Improved background subtraction method for detecting moving objects based on GMM. *IEEJ Transactions on Electrical and Electronic Engineering*, *13*(11), 2018, pp.1540-1550.

[25] Khatri, K. ., & Sharma, D. A. . (2020). ECG Signal Analysis for Heart Disease Detection Based on Sensor Data Analysis with Signal Processing by Deep Learning Architectures. Research Journal of Computer Systems and Engineering, 1(1), 06–10. Retrieved from https://technicaljournals.org/RJCSE/index.php/journal/article/view/11

[26] Tah, A., Roy, S., Das, P. and Mitra, A., Moving object detection and segmentation using background subtraction by kalman filter. *Indian Journal of Science and Technology*, *10*(19) ,2017..

[27] Ali, Y.H. and Mohammed, M.R., 2019. Extracting Background Model in Video Surveillance By Using Hybrid Techniques. *AL-MANSOUR JOURNAL*, (31).

[28] Guo, J., Wang, J., Bai, R., Zhang, Y. and Li, Y., September. A new moving object detection method based on frame-difference and background subtraction. In *IOP conference series: materials science and engineering* (Vol. 242, No. 1, p. 012115). 2017, IOP Publishing.

[29] Xu, Z., Min, B. and Cheung, R.C., A robust background initialization algorithm with superpixel motion detection. *Signal processing: image communication*, *71*, 2019, pp.1-12.

[30] Yin, J., Liu, L., Li, H. and Liu, Q., The infrared moving object detection and security detection related algorithms based on W4 and frame difference. *Infrared Physics & Technology*, *77*, 2016,pp.302-315.

[31] Lee, S.H., Lee, G.C., Yoo, J. and Kwon, S., Wisenetmd: Motion detection using dynamic background region analysis. *Symmetry*, *11*(5), 2019, p.621,pp.1-15.

[32] Camplani, M., Maddalena, L., Alcover, G.M., Petrosino, A. and Salgado, L., September. A benchmarking framework for background subtraction in RGBD videos. In *International Conference on Image Analysis and Processing* 2017, pp. 219-229. Springer, Cham.

[33] Sehairi, K., Fatima, C. and Meunier, J., April. A Benchmark of Motion Detection Algorithms for Static Camera: Application on CDnet 2012 Dataset. In *International Conference on Computer Science and its Applications*, 2018, pp. 235-245. Springer, Cham.

[34] Chakraborty, R., Yang, L., Hauberg, S. and Vemuri, B., Intrinsic grassmann averages for online linear, robust and nonlinear subspace learning. *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[35] Mandal, M., Saxena, P., Vipparthi, S.K. and Murala, S., CANDID: Robust change dynamics and deterministic update policy for dynamic background subtraction. In *2018 24th international conference on pattern recognition (ICPR)* , August 2018, pp. 2468-2473,IEEE.

[36] St-Charles, P.L., Bilodeau, G.A. and Bergevin, R., Subsense: A universal change detection method with local adaptive sensitivity. *IEEE Transactions on Image Processing*, *24*(1), 2014.,pp.359-373.

[37] Ahn, H. and Kang, M., Dynamic background subtraction with masked rpca. *Signal, Image and Video Processing*, *15*(3), 2021,pp.467-474.