Electronic Letters on Computer Vision and Image Analysis 24(2):49-69, 2025

Improving Slow-Moving Object Detection in Complex Environments Using a Feature Pooling Enhanced Encoder-Decoder Model

Upasana Panigrahia[a], Prabodh Kumar Sahoo*[b], Manoj Kumar Panda[c], Ganapati Panda[a]

[a] *Department of Electronics and Communication Engineering,C V Raman Global University, Bhubaneswar, 752054, Odisha, India*
[b] *Department of Mechatronics Engineering, Parul Institute Of Technology, Parul University, Waghodia, Vadodara, 391760, Gujarat, India*
[c] *Department of Electronics and Communication Engineering, GIET University, Gunupur, Rayagada, 765022, Odisha, India*
Received 29th of October, 2024; accepted 18th of July 2025

Abstract

The ability to detect moving objects is of great importance in a wide range of visual surveillance systems, playing a vital role in maintaining security and ensuring effective monitoring. However, the primary aim of such systems is to detect objects in motion and tackle real-world challenges effectively. Despite the existence of numerous methods, there remains room for improvement, particularly in slowly moving video sequences and unfamiliar video environments. In videos where slow-moving objects are confined to a small area, it can cause many traditional methods to fail to detect the entire object. However, an effective solution is the spatial-temporal framework. Additionally, the selection of temporal, spatial, and fusion algorithms is crucial for effectively detecting slow-moving objects. This article presents a notable effort to address the detection of slowly moving objects in challenging videos by leveraging an encoder-decoder architecture incorporating a modified VGG-16 model with a feature pooling framework. Several novel aspects characterize the proposed algorithm: it utilizes a pre-trained modified VGG-16 network as the encoder, employing transfer learning to enhance model efficacy. The encoder is designed with a reduced number of layers and incorporates skip connections to extract essential fine and coarse-scale features crucial for local change detection. The feature pooling framework (FPF) utilizes a combination of different layers including max-pooling, convolutional, and numerous atrous convolutional with varying rates of sampling. This integration enables the preservation of features at different scales with various dimensions, ensuring their representation across a wide range of scales. The decoder network comprises stacked convolutional layers effectively mapping features to image space. The performance of the developed technique is assessed in comparison to various existing methods, including those by CMRM, Hybrid algorithm, Fast valley, EPMCB, and MOD-CVS, showcasing its effectiveness through both subjective and objective analyses. It demonstrates superior performance, with an average F-measure (AF) value of 98.86%, average precision of 98.86%, average recall of 98.87%, and a lower average misclassification error (AMCE) value of 0.85. Furthermore, the algorithm's effectiveness was validated on Imperceptible Video Configuration video setups, where it exhibits superior performance.

Key Words: Background subtraction, Deep neural network, Transfer learning, Slow moving object, Feature pooling framework, Encoder-Decoder type network.

Correspondence to: manojkumarpanda@giet.edu

Recommended for acceptance by Angel D. Sappa
https://doi.org/10.5097/rev/elcvia.2025
ELCVIA ISSN:15108-5097
Published by Computer Vision Center / Universitat Autònoma de Barcelona, Barcelona, Spain

1 Introduction

Visual surveillance plays a crucial role in ensuring safety and typically involves two main steps: foreground extraction and motion tracking [1]. In developing a robust surveillance architecture, the detection of localized changes is essential [2, 3]. Over the past several decades, observing moving object in challenging video environments has become a difficult and actively researched area within visual surveillance systems. Foreground segmentation from image frames finds applications in various domains such as activity recognition [4], traffic monitoring [5], industrial surveillance [6], and underwater monitoring [7]. This process involves differentiating moving objects from the background in complex video sequences, effectively carrying out a binary classification task where background pixels are discarded, and those corresponding to moving objects are preserved. Separating the foreground from complex video scenes is challenging due to dynamic backgrounds, camera movements, missing information, and slow-moving objects. Background subtraction (BGS) approaches [8] have traditionally been used to separate foreground from background, effectively isolating moving objects in image frames. Despite the development of numerous techniques worldwide, current BGS methods tend to perform well only under certain conditions and often depend on manual parameter adjustments and handcrafted features. This highlights the need for more efficient and robust approaches to moving object detection. Deep learning frameworks, which have significantly advanced computer vision applications, are now extensively utilized for medical image analysis [9], Grasping moving object [10], Road detection and monitoring [11], Sustainable development [12], and moving object detection [13, 14, 15, 16] due to their ability to capture low, mid, and high-level features. Moreover, employing transfer learning strategies can further enhance the efficiency of deep neural networks (DNNs) in this regard.

Numerous limitations have been recognized in the DNN designed for moving object detection. Integrating DNNs into visual surveillance systems increases system complexity. It has been observed that as the depth of network layers increases, so does the model's complexity. Moreover, training deep neural networks requires a larger dataset of sample frames. Furthermore, existing techniques rarely feature an end-to-end architecture to detect the objects in motion.

Thus, a notable encoder-decoder model is developed to efficiently tackle various challenges encountered in slow-moving video scenes. To enhance the robustness of the model, the proposed approach utilizes an altered pre-trained VGG-16 DNN as the encoder. The VGG-16 deep neural network [17] is chosen as the encoder for this application due to its proven effectiveness and widespread adoption in various computer vision tasks. Several advantages make the VGG-16 a robust encoder network which includes: VGG-16 has been pre-trained on the ImageNet dataset, which contains over a million images across a thousand classes. This extensive pretraining enables the network to generalize well to a wide range of visual features, making it suitable for transfer learning in various domains. Also, the VGG-16 architecture is simpler compared to other deep networks like ResNet or Inception, with a straightforward structure that is easy to implement and modify. Its standardized architecture, with consistent convolutional filter sizes (3×3) and max-pooling layers, ensures a balance between model complexity and computational efficiency. Further, VGG-16 has been successfully applied in numerous tasks beyond image classification, such as object detection, image segmentation, and even style transfer. Its ability to capture hierarchical features from images has made it a popular choice for applications requiring detailed spatial feature extraction. Furthermore, given its hierarchical feature maps, VGG-16 is particularly effective in feature transfer, where the pre-trained network serves as a robust feature extractor. This quality is beneficial in tasks where computational resources are limited, but high accuracy is desired.

This approach takes advantage of the learned weights of the initial two blocks from the VGG-16 network, while specifically fine-tuning the third block weights on challenging databases. The weights of the initial two blocks of the pre-trained VGG-16 architecture in the BGS model are kept unchanged. However, to tackle the complexities of a challenging dataset, the weights of the third block are updated using transfer learning strategies [1]. By doing so, the model becomes more resilient and better equipped to handle complex data scenarios. Utilizing transfer learning, the modified VGG-16 network retains pertinent features crucial for moving

object detection. The feature maps generated by the encoder are then processed through a feature extraction framework, capturing a variety of details at multiple scales. The decoder framework in the developed scheme efficiently translates feature labels into pixel labels.

Hence, the developed architecture makes four main contributions:

- This study presented an enhanced feature pooling framework (FPF) integrated with the revised VGG-16 architecture, which effectively captures intricate details at various levels, particularly for objects with slow motion.
- The suggested framework is relatively lightweight compared to competitive architectures as it utilizes only three blocks from the VGG-16 architecture.
- Compared to recent existing methods, the suggested approach achieved an average F-measure (AF) of 98.86%, average precision of 98.86%, average recall of 98.87%, and a low average misclassification error (AMCE) of 0.85, using fewer training samples and without relying on temporal information.
- Incorporating a transfer learning process into the developed technique allows the network to effectively learn weights and improve overall performance.

The effectiveness of the presented model is validated using benchmark datasets tailored for slowly-moving object detection, as referenced in [18, 19]. In the proposed method, ten slow-moving object videos are gathered from the dataset. This dataset includes a variety of video sequences characterized by different frame sizes ranging from 144×176 to 360×528 , including a total of 13,606 frames, and content types, ensuring a broad spectrum of testing scenarios. The high-quality, uncompressed format of these videos preserves the original details, facilitating accurate bench-marking and analysis. The developed algorithm's outcomes are compared against five existing methods to corroborate our findings.

We assessed the performance using various metrics such as AF (average F-measure) and AMCE (average misclassification error). We assessed the performance using various metrics such as AF (average F-measure) and AMCE (average misclassification error). The F-measure balances the trade-off between precision and recall, providing a single metric that reflects the quality of the detection in scenarios where both false positives and false negatives are important. A high F-measure indicates that the detection algorithm effectively identifies moving objects (high recall) without mistakenly classifying too many non-moving objects as moving (high precision). This metric is crucial when the focus is on both detecting as many true moving objects as possible while minimizing the number of false alarms. The average misclassification error quantifies the proportion of pixels or objects that are incorrectly classified in the detection process, encompassing both false positives and false negatives. It provides a clear indication of the overall accuracy of the moving object detection system, giving a straightforward measure of how often the algorithm fails to correctly classify the objects. Lower misclassification error signifies a more accurate detection system, which is essential for applications where precision and accuracy are critical. To assess the performance of the developed technique, both subjective and objective assessments were conducted, demonstrating its efficacy.

The subsequent sections of the article are structured as follows: Section 2 outlines the existing literature on moving object detection. Section 3 delves into an in-depth exploration of the proposed model, accompanied by graphical illustrations. Section 4 presents the analysis of empirical results and an ablation study. Finally, Section 5 offers the conclusions drawn from the article.

2 Related Literature

2.1 Existing Methods for Slow Moving BGS

The method of recognizing and monitoring objects that move at a slow pace is called object detection. Generally, the movement of these objects is limited to a small region. Despite significant advancements in object

detection, existing methods still struggle to accurately detect slow-moving objects in complex environments, primarily due to minimal spatial changes and background noise. This study aims to address this challenge by employing a novel VGG-16 architecture enhanced with a feature pooling mechanism to improve detection accuracy in these scenarios. The predominant techniques for detecting slow-moving objects include frame subtraction (FS) [20], optical flow (OF) [21], foreground detection (FD) [22], feature extraction techniques (FE) [23, 24], algorithmic learning methods (AL) [25]. The chosen method depends on the particular application, the nature of the objects moving at slow speed, and the unrestricted computative aids. A combination or modification of different methods might be required to accurately detect and subsequently track slowly moving objects in various situations [26]. The FD method is valuable for identifying swiftly or moderately moving objects in a scene. It gives efficient computational processing, real-time results, and accurate foreground segmentation particularly when objects move slowly against a mostly stationary background. Nevertheless, this method is affected by lighting variations, confined to unchanging backgrounds, encounters challenges with obstruction, and requires background construction if the same is unavailable. Furthermore, it struggles to detect slow-moving objects because of minimal spatial changes in the object's pixel area [27, 28, 29]. The motion vectors of pixels are utilized in the OF method for moving object detection to ascertain the direction and speed of movement. This technique excels in accurately identifying and tracking fast-moving objects. Consequently, it proves highly adaptable to variations in texture, lighting, and other environmental factors, making it optimal for real-time object tracking in video surveillance scenarios. However, it struggles to effectively handle occlusion and is susceptible to image noise. Moreover, it lacks the capability to offer depth details about the tracked object. It may not be effective for stationary or slow moving objects as it dependent on the object movement [21, 30, 31]. FD is a widely-used technique for identifying moving objects within a sequence of images. It provides a swift and real-time method suitable for integration into surveillance systems. By analyzing variations between frames, FD offers a cost-effective solution capable of detecting even partially obscured objects. However, its susceptibility to artifacts and minor disturbances such as sensor instability or variations in illumination may result in false positives, thus potentially compromising the effectiveness of the results. It solely identifies moving objects that contrast with the background, rendering it inadequate to locate objects with a homogeneous appearance to the background [32, 33]. FE methods excel at managing challenging scenarios where an object's appearance undergoes changes due to complex backgrounds or varying lighting conditions. These techniques, designed to extract specific image features, efficiently handle computational demands and can process video streams in real-time. However, their performance significantly declines when robust feature detection is compromised by noise, occlusion, or other factors. Moreover, adapting to new object classes or motion features often requires frequent adjustments or retraining. Notably, these methods prioritize primitive visual features and devoid of advanced semantic understanding. Consequently, they may struggle to distinguish between objects with similar basic characteristics, resulting in false detections or tracking deviations [34, 35, 36, 37]. On the other hand, some of the existing ML techniques have been explored in the literature for the detection of slowly moving objects. Wei Liu $et\ al.$ introduced the Single Shot Multi Box Detector (SSD) object detection algorithm, which effectively achieves a optimal trade-off between precision and speed for object detection in images. SSD creates the final set of object detections by applying non-maximum suppression (NMS) to remove redundant bounding box predictions. The main advantages of SSD are its speed, ease of use, and multi-scale object detection capabilities. However, SSD sacrifices some accuracy in favor of quicker inference times. It detects objects at different scales by employing a predefined set of anchor boxes. However, it can be challenging to select the appropriate sizes and proportions for these anchor boxes. Significant deviations from these predefined anchor boxes could lead to incorrect object identification. It is also unable to handle heavily obscured objects [38]. The introduction of Faster R-CNN by Ren $et\ al.$ significantly impacted computer vision, becoming a widely adopted object detection technique [39]. This approach precisely and efficiently identifies objects by utilizing a region proposal network to create contender object bids. This method allows for end-to-end training of the object detection system, ensuring maximum efficiency. However, it is more intricate than earlier object detection techniques. It includes elements like a shared convolutional network core, a target-specific classifier, and a region proposal network. Its complexity may make it more challenging to comprehend and apply. Focal

Loss is a groundbreaking loss function specifically tailored for detecting densely packed objects applications, including object recognition and individual instance segmentation by Tsung-Yi Lin *et al*. The focal loss suggests a focusing parameter, an extra hyper-parameter, that regulates the rate at which the loss decreases for simple negative instances. Selecting an optimal parameter configuration requires meticulous adjustment, and an incorrect configuration can impact the evaluation of the outlined approach [40]. Corsel *et al*. [41] introduced a spatio-temporal deep learning model, derived from YOLOv5, which harnesses temporal context by analyzing sequences of frames simultaneously. The model substantially enhances the recognition of minuscule moving objects in aerial surveillance and person detection contexts, all the while maintaining the detection accuracy of stationary objects. Despite its effectiveness in enhancing the detection of tiny objects through temporal context, this approach may require significant computational resources and may not be well-suited for real-time applications due to increased processing time. CornerNet, an innovative object detection framework introduced by Hei Law and Jia Deng, treats objects as paired key-points to facilitate detection. In CornerNet, objects are depicted as key-points with their spatial information modeled, resulting in enhanced localization accuracy and decreased false positives. However, this method, which perceives objects as focal points, may encounter challenges when handling objects with intricate or significantly varied poses. Given the model's primary emphasis on corner detection, its effectiveness might be diminished in scenarios where key-points lack prominence or informative [42]. Lee *et al*. [43] proposed Adversarially-trained feature interpolator Generative Adversarial Networks (AFI-GAN). AFI-GAN enhances feature interpolation within Feature Pyramid Networks (FPNs) using adversarial training, resulting in more precise object detection by efficiently managing scale variations. By addressing the limitations of traditional feature interpolation methods, AFI-GAN can potentially improve the accuracy of object detection systems, especially in scenarios with significant scale variations. While AFI-GAN may improve feature interpolation within FPNs, its effectiveness could vary across different datasets or object detection tasks, limiting its generalization capability in diverse settings. Nevertheless, all the aforementioned techniques can only accurately identify objects that are moving swiftly or moderately within the scene. This inspired us to utilize the VGG-16 architecture with structural modifications to develop a framework for detecting slow-moving objects.

3 Proposed Method

This work introduces an innovative cutting-edge deep neural network designed specifically extract foreground regions in challenging scenarios. The model exhibits remarkable resilience and efficiency, making it a robust solution for accurately separating foreground objects from the background in intricate video scenes. The proposed model employs a DNN architecture, in which a modified version of the VGG-16 network functions as the encoder. To accurately identify objects of varying scales within video frames, the model also developed a feature pooling framework (FPF). This combination of advanced techniques enables the model to efficiently detect and process complex visual data, making it an effective solution for a wide range of applications requiring object detection in video content. Additionally, the FPF module is capable of preserving both scattered and concentrated features from image frames, enabling effective local change detection. The decoder network then learns to map feature labels to pixel labels accurately. The comprehensive structure of the proposed network is depicted in Fig. 1.

3.1 Encoder Network

In this study, we utilize a pre-trained VGG-16 network as the encoder network, a commonly employed architecture in various image-processing tasks. However, its potential for foreground segmentation remains unexplored for slow moving object detection. Here, we leverage the capabilities of the VGG-16 network specifically for foreground separation. The VGG-16 architecture comprises five blocks, each of which is equipped with convolutional layers and rectified linear unit (ReLU) functions. This unique combination of components allows the model to preserve spatial details from the input image and selectively boost neuron activity through the ReLU

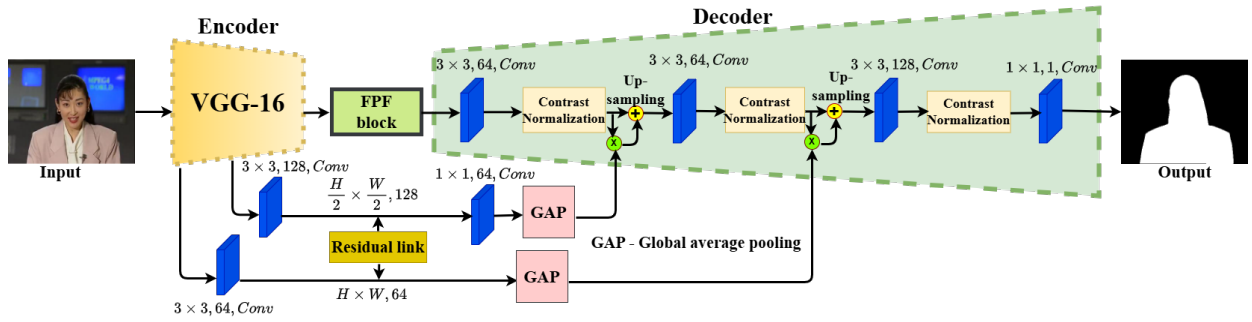

Figure 1: Structural diagram illustrating the proposed BGS model.

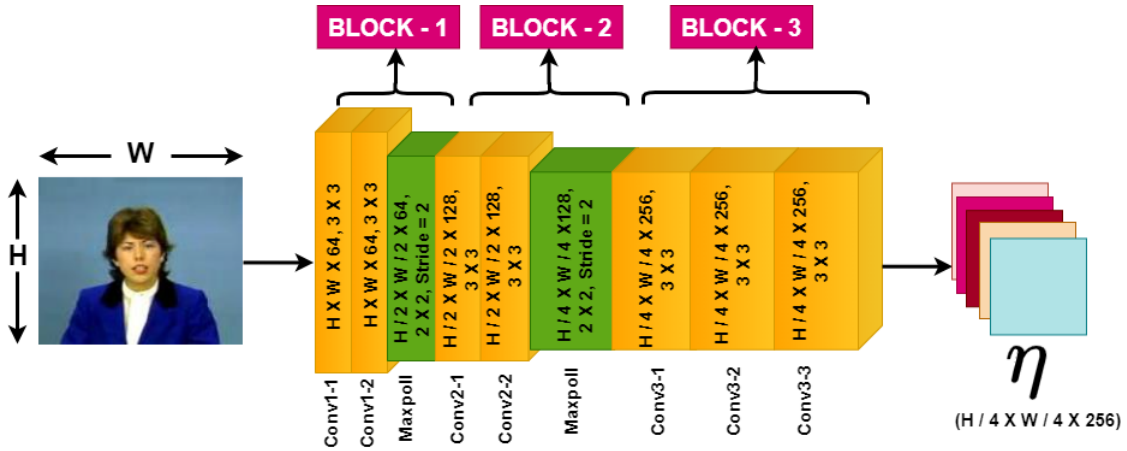

Figure 2: Structural overview of the proposed VGG-16 architecture.

activation function. As a result, the model's overall performance is enhanced, making it a powerful tool for a variety of computer vision applications.

The developed model is built upon a modified version of the deep VGG-16 network, focusing on the initial three blocks as shown in Figure 2. The weights of the initial two blocks of the pre-trained VGG-16 architecture in the BGS model are kept unchanged. However, to tackle the complexities of a challenging dataset, the weights of the third block are updated using transfer learning strategies [1]. This adaptation process enables the model to leverage existing learned features and improve its performance in handling the challenges of the dataset. Also, transfer learning facilitates the transfer of knowledge from one task domain to another, enhancing the model's adaptability and efficiency, particularly when training data is limited. The third block of VGG-16 strikes a balance between low-level and high-level feature extraction. While the initial blocks capture low-level details such as edges and textures, the third block begins to abstract more meaningful patterns like shapes and object parts, which are crucial for downstream tasks such as object detection or segmentation. Fine-tuning only this block allows adaptation to the target dataset without compromising the general feature representations learned in earlier layers or overfitting due to the high specialization of deeper layers. This selective tuning also reduces computational cost while retaining sufficient learning ability for domain-specific improvements. The proposed approach seeks to enhance the use of fine spatial detail and frequency details by excluding the fourth and fifth blocks of the VGG-16 architecture. Within the initial encoder block, 3×3 convolutional layers with 64 and 128 filters are employed to capture fine-scale features [1]. Through a series of comprehensive experiments, it has been consistently observed that utilizing a configuration of (64, 128)

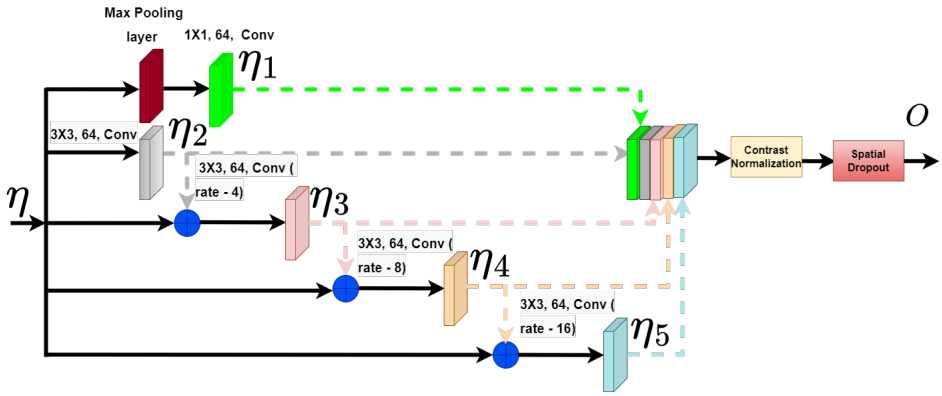

Figure 3: Structural layout of the developed feature pooling framework.

filters in the developed model significantly enhances performance, yielding remarkable AF and AMCE scores of 98.86% and 0.85, respectively. In comparison, employing (16, 32) or (32, 64) filters resulted in lower AF and AMCE values of 93.74% and 0.94, and 97.52% and 0.89, respectively. However, pushing the proposed model with (128, 256) filters can afford a marginal improvement to 98.89% and 0.83 for AF and AMCE scores, albeit at the cost of significantly increased computational resources. Taking these findings into consideration, the (64, 128) filter configuration has been considered for the proposed algorithm. This choice optimally balances high computational efficiency with an exceptional capability to meticulously maintain the fine-scale details in the visual sequences. This decision not only upholds the model's performance but also ensures the preservation of critical nuances from video scenes, making it a preferable choice for real-world applications. Subsequently, to maintain the integrity of the feature representation, the decoder network receives these features through global average pooling (GAP) and residual connections.

3.2 Feature Pooling Framework

In this study, a novel FPF module is introduced, strategically positioned between the encoder and decoder networks, as depicted in Figure 3. The aim is to precisely capture objects of various sizes within challenging slow-moving video scenes. The FPF module incorporates multiple elements, such as a hybrid max-pooling layer paired with a 64-channel convolutional layer featuring a 1×1 filter, a convolutional layer with 64 channels, and a 3×3 filter. It also integrates atrous convolutional layers with sampling rates of 4, 8, and 16. In particular, the method we have developed makes effective use of atrous convolutional layers, employing a 64-channel 3×3 filter size configuration. The inclusion of a max-pooling layer is vital as it helps retain the most critical information, denoted as η_1, by using 2×2 windows from the outcome of the encoder network η. Moreover, the FPF block incorporates multiple layers, including the convolutional layer and atrous convolutional layers whose outcomes are denoted as η_2, η_3, η_4, and η_5. These layers synergistically capture both insubstantial and dense features from the feature space η, resulting in a robust and comprehensive representation of the data. As a result, the features η_1, η_2, η_3, η_4, and η_5 are concatenated across channels and then subjected to contrast normalization. Subsequently, a spatial dropout layer (SDL) is utilized with a 0.25 dropout rate, effectively introducing regularization to the FPF block. This dropout layer contributes to the generation of 240 feature maps, enhancing the overall feature extraction process. The effectiveness of an SDL with a rate of 25% in the proposed algorithm is often found to outperform other values for several key reasons, which are discussed as follows: An SDL with a rate of 0.25 functionally drops out 25% of the feature maps across the entire spatial dimensions for a segment of the input tensor. By doing so, it reduces the correlation among feature maps and encourages the network to learn more diverse and robust feature representations. Setting the rate to 0.25 offers an optimal balance between overfitting and under-fitting. Too low a dropout rate may not impact

co-adaptation and over-fitting issues effectively, while too high a dropout rate might lead to a complete loss of useful information, hindering learning and causing under-fitting. A 25% dropout rate strikes a balance, allowing the network to learn but with enough limitations to prevent it from memorizing training data too closely. At 0.25, the network still maintains 75% of its feature information, ensuring its learning sufficient representations while concurrently mitigating the risk of over-reliance on specific features. This setup allows the network to handle noise and variations in input data better. The choice of 0.25 balances the regularization strength against the computational cost. Lower dropout rates provide less regularization but require more training epochs to achieve the same effect, while higher rates increase the risk of over-regularization without offering proportional benefits. A rate of 0.25 is often found to be practical in terms of achieving efficient regularization without excess computational load. The results of the experiments demonstrate that replacing batch normalization with contrast normalization improves the effectiveness of the model. Moreover, employing SDL adequately preserves spatial information while minimizing redundant data. These findings highlight the effectiveness of these techniques in enhancing the overall performance and efficiency of the developed model.

3.3 Decoder Network

Accurate detection of slow-moving objects within complex video scenes heavily relies on spatial data. To preserve this essential information, the decoder network is based on the designed model as presented in Figure. 1 is built with a series of convolutional layers. This enables the model to effectively capture and preserve the spatial details, thereby improving the accuracy of slow-moving object detection. Initially, the decoder network converts the 240 feature maps obtained from the FPF block into a set of 64 feature maps. This transformation is achieved through the utilization of a convolutional layer that employs 64 filters, each with a size of 3×3. These obtained features then undergo contrast normalization. These feature maps are then merged with the fine-scale features extracted from the final layer of the first encoder block, using the ReLU function and a GAP layer. The inclusion of the GAP layer within the decoder framework significantly contributes to enhancing the overall performance of the model. After the feature fusion, they are subjected to up-sampling and subsequently pass through an additional convolutional layer that consists of 64 numbers of filters, each with a size of 3×3. Subsequently, contrast normalization and ReLU activation are applied, resulting in the generation of 64 features. The feature maps are subsequently merged with the fine-detailed features preserved from the initial block of the encoder, and then the GAP layer is applied. Further enhancement is achieved by up-sampling the fused features and projecting them into 128 feature maps by using a convolutional layer equipped with 128 numbers of filters, each having a size of 3×3. These features significantly improve the representation of objects and background pixels, thereby boosting the model's performance. Finally, a sigmoid activation function and a convolutional layer with a single 1×1 filter map the feature space precisely to the image space. When dealing with complex video scenes, the application of a threshold value of 0.9 yields outstanding results in the creation of masks for the related RGB (Red-Green-Blue) input image.

4 Results and Discussion

The presented framework runs on a Windows 10 OS, RAM of 8GB, using Python programming. The designed approach includes training and testing on the high-performance *NVIDIA* Tesla *T4* GPU, which is made available through the Google Co-lab. The innovative technique utilizes the powerful *TensorFlow* backend alongside the versatile *Keras* library. The efficacy of the introduced model is evaluated on demanding datasets characterized by slow-moving dynamics [18, 19]. We have validated the efficacy of the proposed algorithm by conducting a comprehensive analysis, comparing its results with those achieved by five existing techniques. Furthermore, the developed model is also validated using a larger dataset, CDnet-2014 [47]. This evaluation is carried out using both objective and subjective measures.

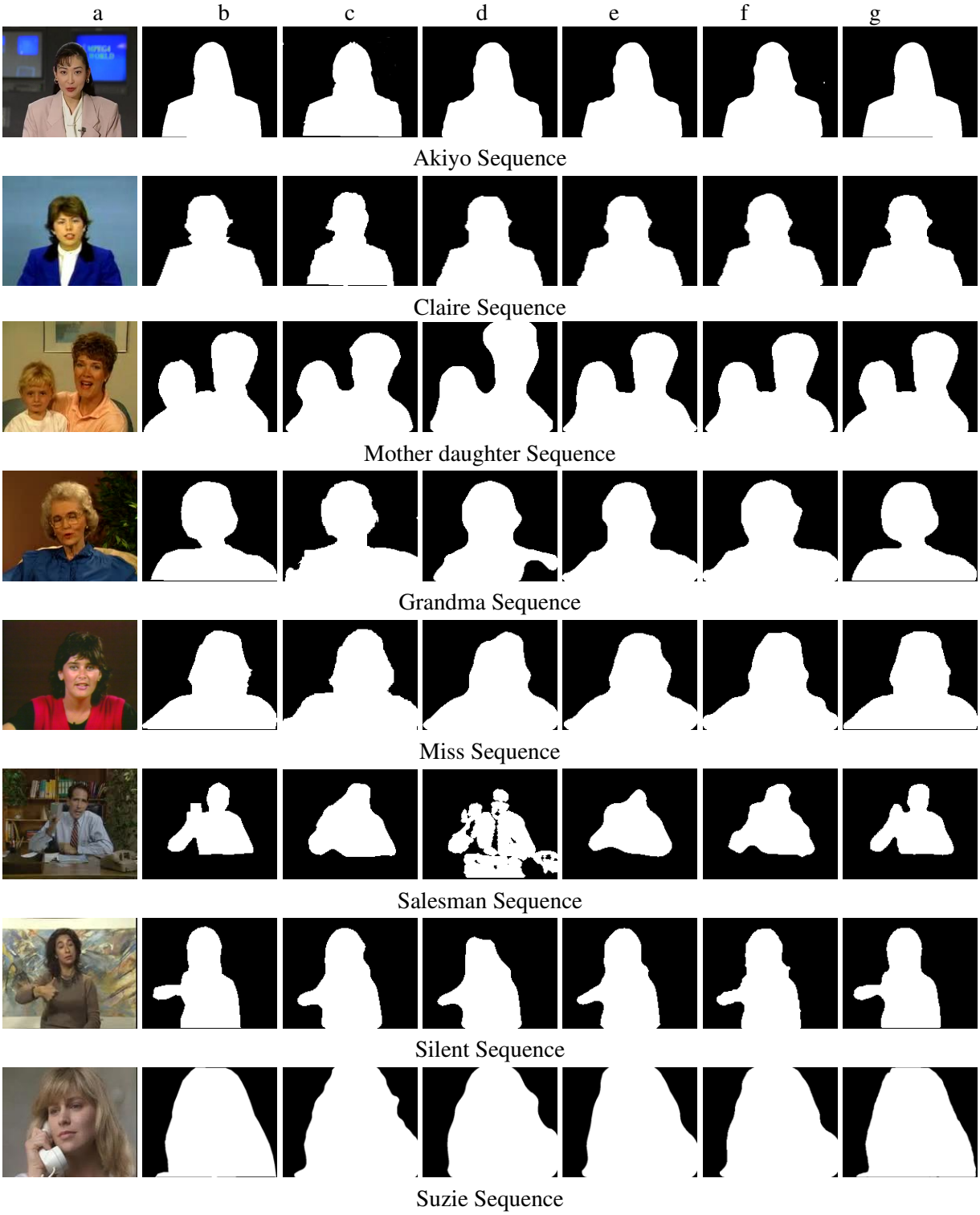

Figure 4: Detection outcomes for diverse testing sequences: (a) input frame (b) ground-truth frame, results acquired by BGS techniques based on (c) CMRM [44], (d) Hybrid algorithm [45], (e) Fast valley [26], (f) EPMCB [46] and (g) Proposed schemes.

Figure 5: Detection outcomes for diverse testing sequences: (a) input frame (b) ground-truth frame, results acquired by BGS techniques based on (c) CMRM [44], (d) Hybrid algorithm [45], (e) Fast valley[26], (f) EPMCB [46] and (g) Proposed schemes.

4.1 Parameter configuration and Training specifics

The designed model is trained using a *NVIDIA* Tesla *T4* GPU setup, batch size equal to 2 being utilized during the training process. The decreased batch size in the developed model may induce a distinct regularization

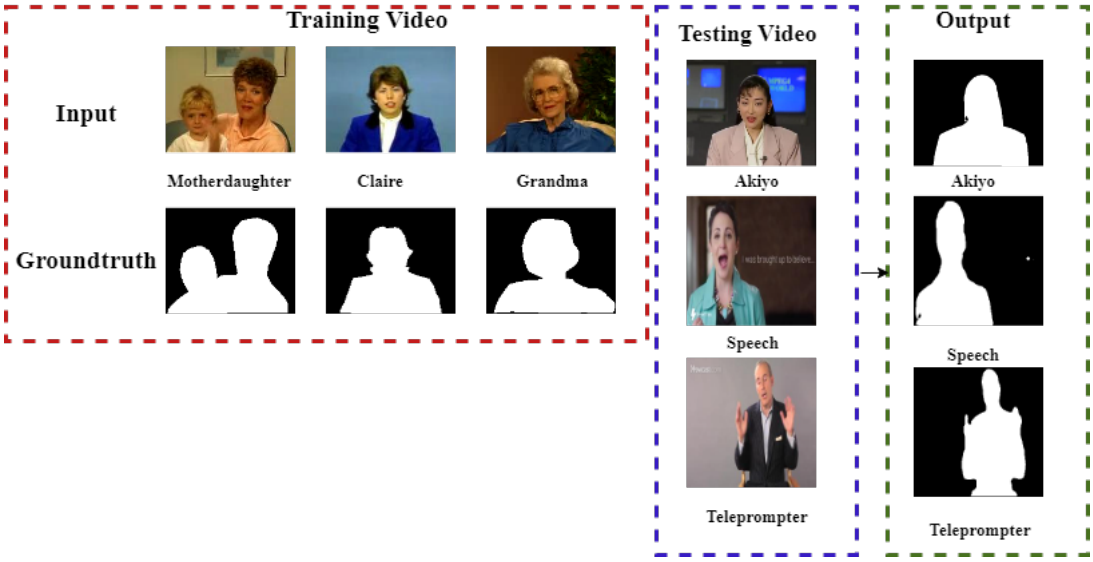

Figure 6: Training and testing samples along with results for imperceptible video setup.

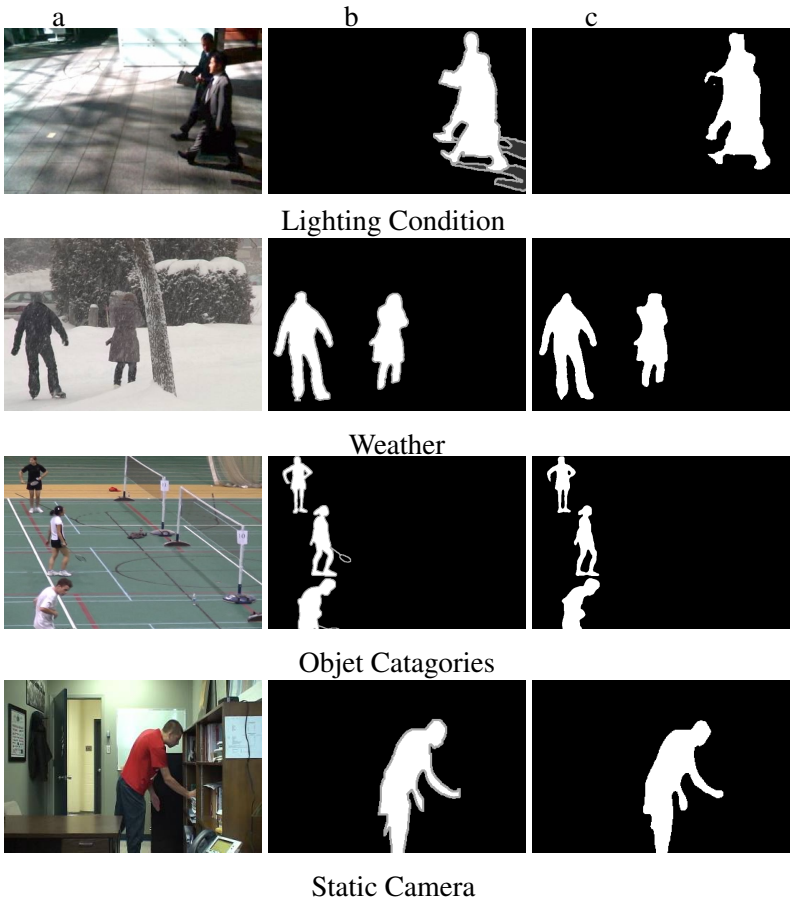

Figure 7: Segmentation of foreground in different sequences: (a) input frame (b) ground-truth frame, (c) results acquired by BGS technique for CDnet-2014 dataset.

Table 1: Specification of the model's parameters

Parameter	Value
Learning rate	0.0001
Batch size	2
ρ	0.9
Maximum epoch	100
ϵ	1e-08

Table 2: Comparative analysis of AF in percentage with baseline methods on slow moving dataset

Sample Video	Methods					
	Hybrid algorithm [45]	Fast valley [26]	CMRM [44]	EPMCB [46]	MOD-CVS [2]	Proposed
Miss	90.77	96.94	90.16	97.14	**99.11**	98.91
Grandma	83.69	95.04	83.24	95.31	98.73	**98.78**
Silent	91.72	95.55	90.83	95.77	97.97	**98.42**
Suzie	88.64	97.50	88.20	97.84	98.54	**98.76**
Akiyo	96.06	98.37	96.01	98.49	99.26	**99.41**
Mother daughter	90.38	96.24	89.21	97.09	**99.25**	99.20
Claire	96.65	98.50	97.03	98.88	98.68	**98.90**
Teleprompter	97.16	98.63	97.23	98.96	98.82	**98.98**
Salesman	74.42	92.77	69.56	93.79	97.94	**98.13**
Speech	58.99	94.46	57.63	96.34	98.83	**99.15**
Average	86.85	96.40	85.91	96.96	98.71	**98.86**

effect, facilitating faster convergence. The model is trained with P pixels per frame over a sequence of frames of N = 25. Moreover, we perform model training using the binary cross-entropy loss function. This loss function helps us assess the classification of individual pixels by comparing the actual class labels with the predicted ones.

To train the proposed methodology, we utilized the *RMSProp* optimizer with specific parameter values. With $\rho = 0.9$ and $\epsilon = 1e - 08$, this optimizer offers faster convergence compared to conventional ones. We set the initial learning rate at 0.0001. If the validation loss does not decrease over 5 consecutive epochs, we decrease the learning rate by a factor of 10. We capped the maximum number of epochs at 100 during model training. However, if the validation loss remains unchanged for 10 consecutive epochs, we activate an early stopping mechanism. The details of the configuration of the parameters of the proposed model are presented in Table 1.

To mitigate biased learning weights caused by the sequential presentation of training frames, which can introduce a high correlation between successive frames, we randomly selected training frames. We allocated 20% of the frames for validation and 80% for training. Additionally, to address imbalanced data classification, we assigned higher weights to the foreground class and lower weights to the background class. For example, the Akiyo dataset comprises 288 frames. Out of these, 25 frames are randomly selected, with 20% designated for validation and 80% for training. Once training and validation are complete, the trained model is tested using the entire set of 288 frames. We prioritized the foreground class by giving it higher weights while assigning lower weights to the background class.

4.2 Subjective Measure

In Figure 4, we present a visual comparison of the outcomes obtained using conventional methods and the developed algorithm. The original frames and their interrelated ground-truth frames are shown in Figures 4

Table 3: Comparative analysis of AMCE with baseline methods on slow moving dataset

Sample Video	Methods					
	Hybrid algorithm [45]	Fast valley [26]	CMRM [44]	EPMCB [46]	MOD-CVS [2]	Proposed
Miss	7.96	2.88	3.04	1.75	**0.83**	1.02
Grandma	13.04	4.38	5.74	4.26	1.07	**1.03**
Silent	4.08	3.19	3.66	2.63	1.09	**1.02**
Suzie	5.68	2.92	4.47	2.92	**1.78**	1.98
Akiyo	2.81	1.40	2.02	1.22	0.54	**0.50**
Mother daughter	12.43	3.58	4.55	2.94	**0.73**	**0.78**
Claire	1.92	0.80	3.23	0.84	**0.81**	0.98
Teleprompter	2.56	0.91	3.47	0.71	**0.69**	0.83
Salesman	12.45	4.15	12.22	2.94	0.90	**0.90**
Speech	2.68	2.24	4.12	2.14	0.67	**0.48**
Average	6.56	2.64	4.65	2.33	0.91	**0.85**

Table 4: Comparison of Average Precision (APre) in percentage with baseline methods on slow moving dataset

Sample Video	Methods			
	Hybrid algorithm [45]	Fast valley [26]	EPMCB [46]	Proposed
Miss	92.46	94.02	94.33	**98.89**
Grandma	80.34	83.38	84.06	**98.76**
Silent	64.68	80.43	80.81	**98.43**
Suzie	95.75	97.86	97.87	**98.76**
Akiyo	88.99	93.87	93.88	**99.42**
Mother daughter	90.80	91.15	91.68	**99.21**
Claire	81.12	93.83	91.84	**98.89**
Teleprompter	83.16	89.18	92.30	**98.97**
Salesman	89.08	91.90	93.43	**98.15**
Speech	80.25	90.84	93.43	**99.13**
Average	84.66	90.68	91.56	**98.86**

(a) and (b) respectively. Figure 4 (c) displays the results obtained using the technique proposed by CMRM [44], revealing instances where background pixels were mistakenly identified as objects pixels across different slow-moving video scenes. The outcomes obtained using the approach developed by Hybrid algorithm [45] are depicted in Figure 4 (d), highlighting a significant number of missed alarms. Figures 4 (e) and (f) showcase the results achieved by the methodologies proposed by Fast valley [26] and EPMCB [46], respectively, both exhibiting a considerable false negative rate. In contrast, Figure 4 (g) displays the outcomes obtained by the designed model, demonstrating precise classification of background and foreground pixels. Moving on to Figure 5, (a) and (b) respectively present the input frames and their interrelated ground-truth frames. Notably, Figure 5 (g) clearly illustrates the impact of the developed method, achieving accurate shapes of moving objects with significantly lower rates of false positives and false negatives compared to the methods CMRM [44], Hybrid algorithm[45], Fast valley [26], and EPMCB [46]. Furthermore, the robustness of the designed model is demonstrated using the benchmark CDnet-2014 dataset. As illustrated in Figure 7, the proposed method is validated in different scenarios such as low light conditions, bad weather, object categories and in static cameras. Figure 7 (c) clearly demonstrates that the proposed algorithm accurately detects the objects with reduced false alarms in the challenging video scenes.

Table 5: Comparison of Average Recall (ARe) in percentage with baseline methods on slow moving dataset

Sample Video	Methods			
	Hybrid algorithm [45]	Fast valley [26]	EPMCB [46]	Proposed
Miss	92.45	93.12	94.31	**98.92**
Grandma	80.35	83.35	84.05	**98.77**
Silent	64.69	80.41	80.83	**98.45**
Suzie	95.74	97.85	97.88	**98.77**
Akiyo	89.00	93.86	94.81	**99.40**
Mother daughter	90.81	91.13	91.66	**99.23**
Claire	81.11	93.84	91.83	**99.00**
Teleprompter	89.05	83.17	92.28	**98.96**
Salesman	89.05	91.91	93.40	**98.17**
Speech	80.23	90.82	93.46	**99.10**
Average	85.24	89.94	91.45	**98.87**

Table 6: Comparative analysis of computational time in second with baseline methods on slow moving dataset

Sample Video	Methods			
	Hybrid algorithm [45]	Fast valley [26]	EPMCB [46]	Proposed
Miss	29.74	8.74	10.20	**7.02**
Grandma	39.31	6.71	11.38	**4.07**
Silent	17.04	6.77	11.29	**3.98**
Suzie	107.85	13.29	15.41	**9.34**
Akiyo	82.22	15.69	24.51	**7.74**
Mother daughter	35.64	7.54	14.13	**6.23**
Claire	16.12	4.53	7.05	**3.97**
Teleprompter	33.45	13.37	21.39	**10.96**
Salesman	25.10	11.38	13.63	**8.27**
Speech	27.53	17.18	45.37	**11.22**
Average	41.39	10.52	17.43	**7.28**

4.3 Objective Measures

To gauge the efficacy of the designed approach, we conducted a thorough quantitative investigation and pitted it against cutting-edge techniques specifically tailored for objects with slow movement. This allowed us to make a comprehensive comparison and assess the performance of the presented methodology. Performance is assessed using the AF and AMCE, as detailed in the Tables 2 and 3 respectively. Our analysis of these tables indicates that the proposed technique outperformed existing methods including CMRM [44], Hybrid algorithm [45], Fast valley [26], EPMCB [46], and MOD-CVS [2]. Specifically, the developed algorithm achieved higher accuracy with an AF value of 98.86%, average precision of 98.86%, average recall of 98.87% and a lower AMCE value of 0.85, indicating its superior performance over the existing methods. Further, the proposed approach outperforms traditional methods in terms of precision and recall, resulting in fewer incorrect detections and a higher rate of true object identification, as presented in Table 4 and Table 5 respectively. The computational effectiveness of the formulated model is further substantiated in Table 6, which presents a comparative analysis of execution times (in seconds) against existing approaches, confirming the designed model's less execution time which may be suitable for real-time applications. Furthermore, as shown in Table 7, the evaluation metrics

Table 7: Comparative summary of average performance metrics across 10 slow-moving datasets and baseline approaches

Approaches	AF (%)	AMCE	APre (%)	ARe (%)
Hybrid algorithm [45]	86.85	6.56	84.66	85.24
Fast valley [26]	96.40	2.64	90.68	89.94
EPMCB [46]	96.96	2.33	91.56	91.45
MOD-CVS [2]	98.71	0.91	98.56	98.81
Proposed	**98.86**	**0.85**	**98.86**	**98.87**

Table 8: Ablation study on slow-moving datasets evaluating AF performance with and without GAP module

Video Name	Model without GAP	Model with GAP
Miss	98.35	**98.91**
Grandma	98.32	**98.78**
Silent	**99.07**	98.42
Suzie	98.31	**98.76**
Akiyo	98.28	**99.41**
Mother daughter	98.13	**99.20**
Claire	98.19	**98.90**
Teleprompter	98.86	**98.98**
Salesman	97.32	**98.13**
Speech	98.02	**99.15**
Average	98.28	**98.86**

are summarized based on average performance across the ten slow-moving data sets analyzed. From Table 7 it may be seen that the proposed algorithm obtained superior performance in AF, APre, and ARe, accompanied by a lower AMCE compared to all the methods, ensuring a better detection of moving objects within the complex scenes.

4.4 Ablation Study

In this section, an ablation study was conducted to assess the impact of various factors on the performance of the proposed change detection system. An ablation analysis was conducted to validate the effectiveness of the proposed method, comparing its performance with and without Global Average Pooling (GAP). From Table 8, it has been observed that the proposed scheme, when using GAP, achieves a higher AF compared to when it is not used. Hence, in the proposed algorithm, we incorporate GAP, which effectively retains spatial information for improved feature representation. Additionally, the efficacy of the proposed method is demonstrated in Table 9. Table 9 shows that the proposed Feature Polling Framework attains better AF than without. Further, An ablation study is conducted to examine how changes in the threshold value affect the AF metric. Table 10 indicates that the selected threshold of 0.9 results in improved performance against 0.7 and 0.8. As presented in Table 11, the choice of sampling rate significantly affects the performance of the designed system. From Table 11, it may be found that the sampling rates of 4, 8, and 16 consistently achieved higher AF values than the alternative sampling rate combinations evaluated. Finally, an ablation study is conducted to determine

Table 9: Ablation study on slow-moving datasets evaluating AF performance with and without FPF module

Video Name	Model without FPF	Model with FPF
Miss	97.92	**98.91**
Grandma	98.57	**98.78**
Silent	**99.14**	98.42
Suzie	97.43	**98.76**
Akiyo	98.71	**99.41**
Mother daughter	98.53	**99.20**
Claire	98.81	**98.90**
Teleprompter	97.99	**98.98**
Salesman	98.01	**98.13**
Speech	98.77	**99.15**
Average	98.38	**98.86**

Table 10: Ablation study evaluating the impact of varying threshold values on AF (%) performance

Name Of the Video	Threshold value 0.7	Threshold value 0.8	Threshold value 0.9(Proposed)
Miss	96.82	95.34	**98.91**
Grandma	97.52	97.31	**98.78**
Silent	97.00	97.34	**98.42**
Suzie	96.41	97.43	**98.76**
Akiyo	95.67	94.66	**99.41**
Mother daughter	94.34	97.76	**99.20**
Claire	97.54	96.81	**98.90**
Teleprompter	96.84	96.91	**98.98**
Salesman	98.04	97.16	**98.13**
Speech	97.39	97.70	**99.15**
Average	96.75	96.82	**98.86**

the optimal parameter values for the developed model, focusing on their impact on the AF metric. For each parameter, we varied its value while keeping others constant (at their proposed optimal settings) to observe its effect. As shown in Table 12, the values of the optimized parameters consistently yield the highest AF performance of 98.86% in all the configurations tested. This highlights the importance of precise parameter adjustment to maximize the effectiveness of the designed model in challenging video scenes.

4.5 Imperceptible Video Configuration

In order to carry out the training and evaluation of the proposed model, we used distinct collections of videos that depicted unfamiliar scenarios. To partition the videos into testing and training sets, we employed the leave-one-video-out method. In this work, for an imperceptible video configuration, three videos are considered for training and another three videos are considered for testing. The training videos in this configuration consist of 2,325 frames and testing videos have 9,701 frames. For training setup a total of 25 frames from the three videos are selected randomly which includes various challenges such as illumination variation, dynamic background, low frame rate and two different objects exhibiting motion at two different times. Similarly, the entire frames of testing videos are considered for evaluation, which possess challenges like flashing light variation in the background, and minor facial movements. Likewise, Teleprompter, Salesman, and Speech image sequences are utilized for training, encompassing challenges such as the slow gestures, the subtle eye or head movements,

Table 11: Ablation study evaluating the impact of varying Sampling Rate (SR) on AF (%) performance

Name Of the Video	SR (2, 4, 8)	SR (8, 16, 32)	SR (4, 8, 16) (Proposed)
Miss	95.78	95.65	**98.91**
Grandma	96.59	96.39	**98.78**
Silent	97.05	96.34	**98.42**
Suzie	94.23	93.43	**98.76**
Akiyo	93.85	94.45	**99.41**
Mother daughter	95.65	94.61	**99.20**
Claire	93.50	91.80	**98.90**
Teleprompter	95.84	95.08	**98.98**
Salesman	97.04	96.16	**98.13**
Speech	95.48	94.72	**99.15**
Average	96.75	94.86	**98.86**

Table 12: Ablation study evaluating impact of varying parameter values on AF value on slow moving dataset

Parameters	Values	AF
Learning rate	0.00001, 0.001, **0.0001**	94.76, 95.01, **98.86**
Batch size	1, 3, **2**	95.21, 91.34, **98.86**
ρ	0.7, 0.8, **0.9**	96.75, 96.82, **98.86**
Maximum epoch	50, 150, **100**	92.61, 93.87, **98.86**
ϵ	$1e-07$, $1e-09$, **1e-08**	91.21, 91.03, **98.86**

and synchronization with audio. the model, which is subsequently tested with Miss and Suzie which includes challenges like lack of significant motion variation between frames. This strategy allowed us to thoroughly evaluate the effectiveness of the proposed model in identifying objects in unfamiliar scenarios. Table 13 reveals that the proposed model achieved a superior average F-measure value in the imperceptible Video Configuration scenario. A sample set of training, testing, and output videos is presented in Figure 6.

5 Conclusions

This work focuses on detecting local changes within video scenes using an encoder-decoder deep learning framework. Specifically, the model targets the identification of slowly moving objects in challenging video environments. To achieve precise feature extraction across various levels, our framework includes an encoder that makes use of the VGG-16 DNN. The integration of transfer learning strategies within the encoder network significantly amplifies the efficacy of the proposed system. In addition, the layers of the VGG-16 deep neural network demonstrate exceptional proficiency in retaining critical features at various levels, including low, mid, and high-level features, which plays a vital role in achieving precise object detection. The inclusion of the details pooling mechanism between the encoder and decoder networks proves to be highly effective in maintaining the integrity of objects with diverse scales in intricate video sequences. The algorithm we propose utilizes the FPM model to facilitate the translation from a higher-dimensional feature space to a feature space that encompasses multiple scales and dimensions. This transformation enables us to directly classify pixels into foreground and background categories, with clear and well-defined decision boundaries. Furthermore, the decoder network incorporated in the model we have developed consists of a sequence of convolutional layers, handily transforming the feature space into the visual domain of images. Overall, our approach facilitates the preservation of objects at different scales and enables precise classification of foreground and background pixels, which is crucial for accurate moving object detection.

Table 13: Average F-measure of the designed scheme in imperceptible video configuration on slow moving object dataset

Name of the Video	Proposed scheme
Akiyo	0.87
Miss	0.78
Teleprompter	0.95
Suzie	0.94
Speech	0.79

By conducting both subjective and objective analyses, we have successfully validated the efficacy of the algorithm we have developed. This assessment involved comparing our algorithm against four existing methods, further affirming its effectiveness. The results demonstrate that our model excels in accurately preserving the contours of moving objects while significantly reducing the occurrence of unwanted pores and holes, outperforming existing methods in this regard. The newly developed algorithm attained an AF value of 98.86%, APre of 98.86%, ARe of 98.86% and an AMCE value of 0.85, showcasing its superior accuracy and performance over existing methods. Moreover, the developed technique demonstrates satisfactory performance on unseen video setups. Nevertheless, the proposed approach shows reduced performance when detecting small-sized moving objects, and its effectiveness is particularly pronounced in scenes with dynamic background. Recognizing the significance of the proposed approach, we plan to explore the integration of attention mechanisms or multi-scale context aggregation modules, which can enhance feature representation for small and partially visible objects. In addition, our goal is to develop a more robust hybridized deep neural architecture in future research to further improve detection accuracy.

References

[1] M. K. Panda, B. N. Subudhi, T. Veerakumar, V. Jakhetiya, "Modified resnet-152 network with hybrid pyramidal pooling for local change de- tection", *IEEE Transactions on Artificial Intelligence* 5 (4) (2023) 1599 –1612, https://doi.org/10.1109/TAI.2023.3299903.

[2] P. K. Sahoo, M. K. Panda, U. Panigrahi, G. Panda, P. Jain, M. S. Islam, M. T. Islam, "An improved VGG-19 network induced enhanced feature pooling for precise moving object detection in complex video scenes", *IEEE Access* 12 (2024) 45847 – 45864, https://doi.org/10.1109/ACCESS.2024.3381612.

[3] U. Panigrahi, P. K. Sahoo, M. K. Panda, G. Panda, "A ResNet-101 deep learning framework induced transfer learning strategy for moving object detection", *Image and Vision Computing* 146 (2024) 105021, https://doi.org/10.1016/j.imavis.2024.105021.

[4] R. Poppe, "A survey on vision-based human action recognition", *Image and Vision Computing* 28 (6) (2010) 976–990, https://doi.org/10.1016/j.imavis.2009.11.014.

[5] J. Hsieh, S. Yu, Y. Chen, W. Hu, "Automatic traffic surveillance system for vehicle tracking and classification", *IEEE Transactions on Intelligent Transportation Systems* 7 (2) (2006) 175–187, https://doi.org/10.1109/TITS.2006.874722.

[6] W. Hu, T. Tan, L. Wang, S. Maybank, "A survey on visual surveillance of object motion and behaviors", *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 34 (3) (2004) 334–352, https://doi.org/10.1109/TSMCC.2004.829274.*

[7] K. Rout, B. N. Subudhi, T. Veerakumar, S. Chaudhury, "Spatio-contextual Gaussian mixture model for local change detection in under- water video", Expert Systems with Applications 97 (2018) 117–136, https://doi.org/10.1016/j.eswa.2017.12.009.

[8] B. N. Subudhi, M. K. Panda, T. Veerakumar, V. Jakhetiya, S. Esakkira-jan, "Kernel-induced possibilistic fuzzy associate background subtraction for video scene", *IEEE Transactions on Computational Social Systems 10 (3) (2022) 1314 – 1325, https://doi.org/10.1109/TCSS.2021.3137306.*

[9] S. Sreelakshmi, G. Malu, E. Sherly, R. Mathew, "M-Net: An encoder-decoder architecture for medical image analysis using ensemble learning", Results in Engineering 17 (2023) 100927, https://doi.org/10.1016/j.rineng.2023.100927.

[10] T.-T. Nguyen, C. V. Duy, "Grasping moving objects with incomplete information in a low-cost robot production line using contour matching based on the hu moments", *Results in Engineering 23 (2024) 102414, https://doi.org/10.1016/j.rineng.2024.102414.*

[11] H. Ranjbar, P. Forsythe, A. A. F. Fini, M. Maghrebi, T. S. Waller, "Addressing practical challenge of using autopilot drone for asphalt surface monitoring: Road detection, segmentation, and following, Results in Engineering 18 (2023) 101130, https://doi.org/10.1016/j.rineng.2023.101130.

[12] Y. Lai, "Optimization of urban and rural ecological spatial planning based on deep learning under the concept of sustainable development", *Results in Engineering 19 (2023) 101343, https://doi.org/10.1016/j.rineng.2023.101343.*

[13] T. Bouwmans, S. Javed, M. Sultana, S. K. Jung, "Deep neural network concepts for background subtraction: A systematic review and compar- ative evaluation", Neural Networks 117 (2019) 8–66, https://doi.org/10.1016/j.neunet.2019.04.024.

[14] M. K. Panda, A. Sharma, V. Bajpai, B. N. Subudhi, V. Thangaraj, V. Jakhetiya, "Encoder and decoder network with resnet-50 and global average feature pooling for local change detection", *Computer Vision and Image Understanding 222 (2022) 103501, https://doi.org/10.1016/j.cviu.2022.103501.*

[15] M. K. Panda, B. N. Subudhi, T. Bouwmans, V. Jakheytiya, T. Veerakumar, "An end to end encoder-decoder network with multi-scale feature pulling for detecting local changes from video scene", in: 2022 18th IEEE International Conference on Advanced Video and Signal Based Surveil-lance (AVSS), 2022, pp. 1–8, https://doi.org/10.1109/AVSS56176.2022.9959141.

[16] S. Pavithra, et al., "An efficient approach to detect and segment underwater images using swin transformer", *Results in Engineering 23 (2024) 102460, https://doi.org/10.1016/j.rineng.2024.102460.*

[17] K. Simonyan, A. Zisserman, "Very deep convolutional networks for large-scale image recognition", arXiv preprint arXiv:1409.1556 (2014), https://doi.org/10.48550/arXiv.1409.1556.

[18] M. Reisslein, L. Karam, P. Seeling, F. Fitzek, Yuv video sequences, [Accessed:2016] (2000). http://trace.eas.asu.edu/yuv/.

[19] C. Montgomery, Xiph.org video test media [derf's collection], [Accessed:2016] (2004), https://media.xiph.org/video/derf/.

[20] P. K. Sahoo, P. Kanungo, K. Parvathi, "Three frame based adaptive background subtraction", in: Proceedings of the International Conferenceon High Performance Computing and Applications, 2014, pp. 1–5, https://doi.org/10.1109/ICHPCA.2014.7045375.

[21] J. H. Duncan, T.-C. Chou, "On the detection of motion and the computation of optical flow", *IEEE Transactions on Pattern Analysis & Machine Intelligence 14 (03) (1992) 346–352, https://doi.ieeecomputersociety.org/10.1109/34.120329.*

[22] S. K. Choudhury, P. K. Sa, S. Bakshi, B. Majhi, "An evaluation of background subtraction for object detection vis-a-vis mitigating challenging scenarios", IEEE Access 4 (2016) 6133–6150, https://doi.org/10.1109/ACCESS.2016.2608847.

[23] P. Viola, M. Jones, "Rapid object detection using a boosted cascade of simple features", in: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 1, 2001, pp. I–511–I–518, https://doi.org/10.1109/CVPR.2001.990517.*

[24] P. Dollar, C. Wojek, B. Schiele, P. Perona, "Pedestrian detection: An evaluation of the state of the art", IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (4) (2012) 743–761, https://doi.org/10.1109/TPAMI.2011.155.

[25] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, "You only look once: Unified, real-time object detection, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788, https://doi.org/10.1109/CVPR.2016.91.*

[26] P. K. Sahoo, P. Kanungo, S. Mishra, "A fast valley-based segmentation for detection of slowly moving objects, Signal", Image and Video Processing 12 (2018) 1265–1272, https://doi.org/10.1007/s11760-018-1278-9.

[27] R. Cucchiara, C. Grana, M. Piccardi, A. Prati, "Detecting moving objects, ghosts, and shadows in video streams", *IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (10) (2003) 1337–1342, https://doi.org/10.1109/TPAMI.2003.1233909.*

*[28] P. Kanungo, A. Narayan, P. Sahoo, S. Mishra, "Neighborhood based codebook model for moving object segmentation", in:*Proceedings of the 2nd International Conference on Man and Machine Interfacing, 2017, pp. 1–6, https://doi.org/10.1109/MAMI.2017.8308009.

[29] P. KaewTraKulPong, R. Bowden, "An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection", *Springer US, Boston, MA, 2002, pp. 135–144, https://doi.org/10.1007/978-1-4615-0913-4_11.*

[30] J. Huang, W. Zou, Z. Zhu, J. Zhu, "An efficient optical flow based motion detection method for non-stationary scenes", in: Proceedings of the Chinese Control And Decision Conference, 2019, pp. 5272–5277, https://doi.org/10.1109/CCDC.2019.8833206.

[31] L. Fan, T. Zhang, W. Du, "Optical-flow-based framework to boost video object detection performance with object enhancement", *Expert Systems with Applications 170 (2021) 1–8, https://doi.org/10.1016/j.eswa.2020.114544.*

*[32] J. Guo, J. Wang, R. Bai, Y. Zhang, Y. Li, "A new moving object detection method based on frame-difference and background subtraction",*IOP Conference Series: Materials Science and Engineering 242 (1) (2017) 012115, https://dx.doi.org/10.1088/1757-899X/242/1/012115.

[33] S. S. Sengar, S. Mukhopadhyay," Moving object detection based on frame difference and w4, Signal", *Image and Video Processing 11 (2017) 1357–1364, https://doi.org/10.1007/s11760-017-1093-8.*

[34] J.-D. Shi, J.-Z. Wang, "Moving objects detection and tracking in dynamic scene", Transactions of Beijing institute of Technology, 29 (10) (2009) 858–861.

[35] X. Huang, F. Wu, P. Huang, "Moving-object detection based on sparse representation and dictionary learning", AASRI Procedia 1 (2012) 492– 497, *aASRI Conference on Computational intelligence and Bioinformatics, https://doi.org/10.1016/j.aasri.2012.06.077.*

[36] M. Sava¸s, H. Demirel, B. Erkal, "Moving object detection using an adaptive background subtraction method based on block-based structure indynamic scene", Optik 168 (2018) 605–618, https://doi.org/10.1016/j.ijleo.2018.04.047.

[37] Q. Zhang, T. Xiao, N. Huang, D. Zhang, J. Han, "Revisiting feature fusion for rgb-t salient object detection", IEEE Transactions on Circuits and Systems for Video Technology 31 (5) (2021) 1804–1818, https://doi.org/10.1109/TCSVT.2020.3014663.

[38] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-a. Fu, A. C.Berg, Ssd: Single shot multibox detector, in: B. Leibe, J. Matas, 25 N. Sebe, M. Welling (Eds.), Computer Vision – ECCV 2016, Springer International Publishing, Cham, 2016, pp. 21–37, https://doi.org/10.1007/978-3-319-46448-0_2.

[39] S. Ren, K. He, R. Girshick, J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks (2016)". arXiv:1506.01497, https://doi.org/10.1109/TPAMI.2016.2577031.

[40] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollar, "Focal loss for dense object detection", in: *Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2980–2988, https://doi.org/10.1109/ICCV.2017.324* .

[41] C. W. Corsel, M. van Lier, L. Kampmeijer, N. Boehrer, E. M. Bakker, "Exploiting temporal context for tiny object detection", in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2023, pp. 79–89, https://doi.org/10.1109/WACVW58289.2023.00013.

[42] H. Law, J. Deng, Cornernet: Detecting objects as paired keypoints, in: *Proceedings of the European conference on computer vision (ECCV), 2018, pp. 734–750, https://doi.org/10.1007/978-3-030-01264-9_45.*

[43] S.-H. Lee, S.-H. Bae, Afi-gan: "Improving feature interpolation of feature pyramid networks via adversarial training for object detection", Pattern Recognition 138 (2023) 109365, https://doi.org/10.1016/j.patcog.2023.109365.

[44] B. N. Subudhi, P. K. Nanda, "Detection of slow moving video objects using compound markov random field model", in: *TENCON 2008-2008 IEEE Region 10 Conference, 2008, pp. 1–6, https://doi.org/10.1109/TENCON.2008.4766385.*

[45] Z. Zhu, Y. Wang, "A hybrid algorithm for automatic segmentation of slowly moving objects", AEU-International Journal of Electronics and Communications 66 (3) (2012) 249–254, https://doi.org/10.1016/j.aeue.2011.07.009.

[46] P. K. Sahoo, P. Kanungo, S. Mishra, B. P. Mohanty, "Entropy feature and peak-means clustering based slowly moving object detection in head and shoulder video sequences", *Journal of King Saud University-Computer and Information Sciences 34 (8) (2022) 5296–5304, https://doi.org/10.1016/j.jksuci.2020.12.019.*

[47] Y. Wang, P.-M. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, P. Ishwar, "CDnet 2014: An expanded change detection benchmark dataset", IEEE conference on computer vision and pattern recognition workshops 387–394, 2014, https://doi.org/10.1109/CVPRW.2014.126.