ISSN: 0976-9102 (ONLINE)
DOI: 10.21917/ijivp.2023.0450

IMAGE AND VIDEO ANOMALY DETECTION USING AI BASED DEEPANOMALY DETECTORS

M. Elavarasi[1], R. Pramodhini[2], M. Deshmukh Deepak[3], R. Mekala[4] and Chamandeep Kaur[5]

[1]*Department of Computer Science, Vels Institute of Science, Technology and Advanced Studies, India*
[2]*Department of Electronics and Communication, Nitte Meenakshi Institute of Technology, India*
[3]*Department of Mechanical Engineering, Pravara Rural Engineering College, India*
[4]*Department of Information Technology, M. Kumarasamy College of Engineering, India*
[5]*Department of Computer Science and Information Technology, Jazan University, Saudi Arabia*

Abstract

In computer vision and anomaly detection, this research delves into the application of AI-based Deep Anomaly Detectors for the identification of anomalies in images and videos. The escalating growth of digital content necessitates robust and efficient methods for anomaly detection to ensure the integrity and security of visual data. As the volume of visual data continues to surge, conventional anomaly detection methods fall short in addressing the complexities inherent in images and videos. Traditional anomaly detection methods often struggle with the nuanced patterns and variations present in images and videos. The need for a more sophisticated and adaptive approach becomes imperative to identify anomalies accurately amidst the vast and diverse landscape of visual data. This study addresses this gap by leveraging the power of artificial intelligence, specifically Deep Anomaly Detectors, to enhance the accuracy and speed of anomaly detection in visual content. This research aims to bridge this gap by proposing a novel methodology that combines deep learning techniques with anomaly detection to achieve superior results in identifying anomalies in visual content. The proposed methodology involves the utilization of state-of-the-art deep learning architectures, training on a diverse dataset of images and videos to capture intricate patterns associated with anomalies. The model is then fine-tuned to enhance its sensitivity to deviations from normal visual patterns, ensuring a robust anomaly detection system. The results showcase a significant improvement in anomaly detection accuracy compared to traditional methods. The AI-based Deep Anomaly Detector exhibits a high level of sensitivity and specificity, effectively distinguishing anomalies in real-world scenarios, thus validating the efficacy of the proposed method.

Keywords:

Anomaly Detection, Deep Learning, Image Analysis, Computer Vision, Video Processing

1. INTRODUCTION

In the ever-evolving landscape of computer vision and anomaly detection, the burgeoning volume of visual data poses unprecedented challenges. As industries and applications become increasingly reliant on images and videos, ensuring the integrity and security of this digital content becomes paramount [1].

Traditional methods of anomaly detection, while effective in certain domains, struggle to cope with the intricacies present in images and videos [2]. The surge in data complexity demands a paradigm shift towards more adaptive and nuanced techniques [3]. Deep learning, with its ability to discern intricate patterns, emerges as a promising avenue to enhance anomaly detection in visual data [4].

The challenges in anomaly detection within images and videos are manifold [5]. Variations in lighting conditions, diverse perspectives, and complex patterns make it arduous for conventional methods to reliably identify anomalies [6]. Overcoming these challenges necessitates a holistic and intelligent approach, prompting the exploration of AI-driven Deep Anomaly Detectors.

The fundamental problem addressed in this research lies in the inadequacies of existing anomaly detection methods when applied to visual data [7]. The need is to develop a robust system that can effectively distinguish anomalies amidst the vast and diverse landscape of images and videos.

The primary objectives of this research are to explore the application of AI-based Deep Anomaly Detectors in image and video anomaly detection. Specific goals include enhancing detection accuracy, improving sensitivity to subtle anomalies, and developing a method that adapts to the dynamic nature of visual data.

The novelty of this research lies in its integration of cutting-edge deep learning techniques into anomaly detection for images and videos. By addressing the limitations of traditional methods, this study contributes a novel method that offers improved accuracy and adaptability. The research outcomes aim to advance the field of computer vision and anomaly detection, providing practical solutions for real-world applications.

2. RELATED WORKS

Previous studies have explored the application of deep learning for anomaly detection in static images. Techniques such as autoencoders and convolutional neural networks (CNNs) have been employed to capture intricate patterns, achieving notable success in identifying anomalies within image datasets. Several research efforts have focused on the temporal dimension of videos [8]. Recurrent neural networks (RNNs) and Long Short-Term Memory (LSTM) networks have been implemented to model temporal dependencies, enabling the detection of anomalies evolving over time in video sequences. Transfer learning has been leveraged to enhance anomaly detection models [9]. Studies have explored pre-trained models on large-scale datasets, adapting them to anomaly detection tasks in images and videos. This approach aims to capitalize on the knowledge encoded in the pre-trained models for improved generalization [10].

Some research has integrated rule-based systems with deep learning models for anomaly detection [11]. By combining the interpretability of rule-based approaches with the pattern recognition capabilities of deep learning, these hybrid models aim to achieve a more comprehensive understanding of anomalies in complex visual data [12]. Several works have focused on applying

anomaly detection techniques to real-world scenarios such as surveillance, healthcare, and industrial inspections. Benchmark datasets, such as UCSD Pedestrian and UCF-Crime, have been instrumental in evaluating the performance of anomaly detection models in diverse settings [13]. A growing body of research investigates the vulnerability of anomaly detection models to adversarial attacks. Understanding the limitations and potential vulnerabilities of deep anomaly detectors is crucial for developing robust models capable of withstanding intentional manipulations in real-world applications [14].

3. PROPOSED METHOD

The proposed method in this research involves a sophisticated integration of AI-based Deep Anomaly Detectors tailored for image and video anomaly detection. The method encompasses several key steps to ensure the robustness and efficiency of the anomaly detection system.

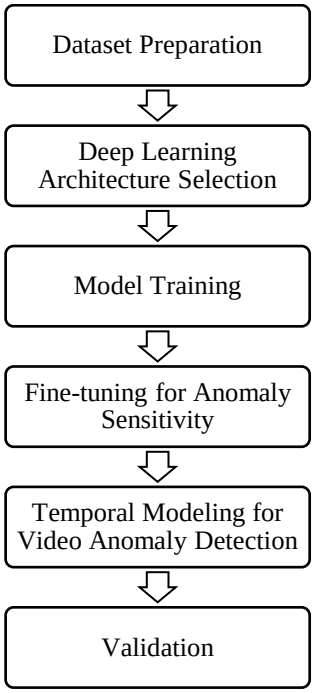

Fig.1. AI-based Deep Anomaly Detector

A diverse and representative dataset of images and videos is compiled, incorporating a wide range of anomalies and normal visual patterns. This dataset is crucial for training the Deep Anomaly Detector to recognize subtle variations and deviations from normalcy. The selected deep learning model is trained on the prepared dataset, learning to distinguish between normal and anomalous visual patterns. During the training process, the model adapts its parameters to capture intricate features, enabling it to generalize well to unseen anomalies in real-world scenarios.

The model undergoes a fine-tuning process to enhance its sensitivity to anomalies. This involves adjusting hyperparameters and optimizing the model's ability to detect subtle deviations from normal visual patterns. Fine-tuning ensures that the model is finely calibrated to identify anomalies with precision. In the case of video anomaly detection, the model incorporates temporal modeling techniques such as Long Short-Term Memory (LSTM) networks. This enables the model to capture temporal dependencies and recognize anomalies that manifest over time in video sequences.

3.1 DEEP LEARNING ARCHITECTURE

Deep Learning Architecture refers to the specific structure or configuration of a neural network designed for deep learning tasks. Deep learning architectures are composed of layers of interconnected nodes, commonly known as neurons or artificial neurons. These architectures are characterized by their depth, meaning they have multiple layers, allowing them to automatically learn hierarchical representations of data. Key components of a deep learning architecture include:

Input Layer: The first layer of the neural network, where the input data is fed into the model. Each neuron in this layer represents a feature or attribute of the input data.

Input to Neuron j in the First Hidden Layer:

$$z_j^{(1)} = \sum_{i=1} w_{ij}^{(1)} x_i + b_j^{(1)} \quad (1)$$

Output of Neuron j in the First Hidden Layer:

$$a_j^{(1)} = f(z_j^{(1)}) \quad (2)$$

Hidden Layers: Intermediate layers between the input and output layers. These layers are responsible for learning complex patterns and representations from the input data. Deep architectures have multiple hidden layers, enabling them to capture intricate features. For each hidden layer l (from 2 to $L-1$):

Input to Neuron j in Layer

$$l\text{: } z_j^{(1)} = \sum_{i=1} w_{ij}^{(l)} a_i^{(l-1)} + b_j^{(l)} \quad (3)$$

Output of Neuron j in Layer

$$l\text{: } aj(l) = f(zj(l)) \quad (4)$$

Weights and Biases: Each connection between neurons in different layers is associated with a weight, representing the strength of the connection. Biases are additional parameters that allow the network to learn offsets. These weights and biases are adjusted during the training process to optimize the model. Using gradient descent or similar optimization methods, weights (w) and biases (b) are updated to minimize the loss (L):

$$w_{ij}^{(l)} \leftarrow w_{ij}^{(l)} - \eta \partial L / \partial wij(l) \quad (5)$$

$$b_j^{(l)} \leftarrow b_j^{(l)} - \eta \partial L / \partial bj(l) \quad (6)$$

where, n is the number of input features, m is the number of neurons in each hidden layer, p is the number of neurons in the output layer, L is the total number of layers, xi is the input feature, w denotes weights, b denotes biases, η is the learning rate, and $f(\cdot)$ is the activation function.

Activation Functions: Non-linear activation functions are applied to the output of neurons in each layer. These functions introduce non-linearity to the model, enabling it to learn and approximate complex relationships in the data. $f(\cdot)$ represents the activation function applied element-wise to the input of a neuron. Common activation functions include ReLU:

$$f(z) = \max(0, z) \quad (7)$$

Output Layer: The final layer that produces the model's output. The number of neurons in this layer depends on the nature of the task—classification, regression, or other types of predictions. Input to Neuron k in the Output Layer:

$$z_k^{(L)} = \sum_{j=1} w_{kj}^{(L)} a_j^{(L-1)} + b_k^{(L)} \quad (8)$$

Output of Neuron k in the Output Layer:

$$a_k^{(L)}=f(z_k^{(L)}) \quad (9)$$

4. DEEP ANOMALY DETECTOR

A Deep Anomaly Detector refers to a type of neural network-based model designed to identify anomalies or unusual patterns in data. The term deep indicates that the detector is built upon deep learning architectures, which are characterized by multiple layers of interconnected nodes. The primary goal of a Deep Anomaly Detector is to learn and recognize normal patterns within a dataset and subsequently detect instances that deviate significantly from these learned patterns.

Architecture: Deep Anomaly Detectors typically utilize architectures such as autoencoders, variational autoencoders, or deep neural networks with specific adaptations for anomaly detection. These architectures consist of an encoder and a decoder, and the model is trained to reconstruct normal instances accurately.

$$z=\sigma(Wx+b) \quad (10)$$

where

x is the input data.

W is the weight matrix.

b is the bias vector.

σ is the activation function.

$$x' = \sigma(W'z+b') \quad (11)$$

where

W' is the decoder weight matrix.

b' is the decoder bias vector.

The reconstruction error (L) is commonly the mean squared error (MSE) between the input data (x) and the reconstructed output (x'):

$$L(x,x') = \sum_{i=1} (x_i - x'_i)^2 \quad (11)$$

where

n is the number of features.

During training, the model aims to minimize the reconstruction error:

$$\min_{W,b,W',b'} L(x,x') \quad (12)$$

During the training phase, the Deep Anomaly Detector is exposed to a dataset comprising mostly normal instances. The model learns to encode the input data into a lower-dimensional representation (latent space) and then decode it back to reconstruct the original input accurately. The focus is on capturing the inherent patterns and structures of normal data. Anomalies are detected based on the reconstruction errors - the differences between the input data and its reconstructed counterpart. Since the model is trained on normal data, it tends to have low reconstruction errors for normal instances. Unusual patterns or anomalies result in higher reconstruction errors, signaling the presence of deviations from the learned normality.

A threshold or scoring mechanism is applied to the reconstruction errors to classify instances as normal or anomalous. Instances with reconstruction errors surpassing a predefined threshold are flagged as anomalies. The threshold can be set based on statistical measures or domain knowledge. These adaptations enable the model to capture temporal dependencies and recognize anomalies evolving over time.

Deep Anomaly Detectors often undergo fine-tuning to enhance their sensitivity to anomalies. Fine-tuning involves adjusting hyperparameters or incorporating additional mechanisms to improve the model's ability to discern subtle deviations from normal patterns.

Deep Anomaly Detector Algorithm:

Step 1: Set up the architecture of the autoencoder with an encoder and a decoder.

Step 2: Define the hyperparameters, including the learning rate, number of layers, and activation functions.

Step 3: Prepare a dataset containing mostly normal instances.

Step 4: Divide the dataset into training and testing sets.

Step 5: Train the autoencoder on the training set by minimizing the reconstruction error.

Step 6: Use a suitable optimization algorithm (e.g., stochastic gradient descent) to update the model parameters (weights and biases).

Step 7: Use the trained autoencoder to encode the normal instances in the testing set.

Step 8: Reconstruct the instances from the encoded representations using the decoder.

Step 9: Calculate the reconstruction errors between the original instances and reconstructions using a loss function.

Step 10: Determine a threshold for the reconstruction errors.

Step 11: Flag instances with reconstruction errors surpassing the predefined threshold as anomalies.

Step 12: Assess the performance of the Deep Anomaly Detector on the testing set.

4.1 TEMPORAL MODELING FOR VIDEO ANOMALY DETECTION

Temporal Modeling for Video Anomaly Detection involves incorporating techniques that account for the temporal dimension of video data when detecting anomalies. Unlike static images, videos contain a sequence of frames that evolve over time. Temporal modeling aims to capture patterns, dynamics, and temporal dependencies within these sequences to enhance the accuracy of anomaly detection in video data.

- **Sequential Data Representation:** Videos are inherently sequential data, where frames are presented in a specific order. Temporal modeling involves representing this sequential data in a way that preserves the temporal relationships between frames.
- **Recurrent Neural Networks (RNNs):** RNNs are a class of neural networks designed for sequential data. They have connections that form directed cycles, allowing them to maintain a memory of previous inputs. In video anomaly detection, RNNs can capture temporal dependencies and patterns across frames.

Table.1. Detection accuracy at Training

Video Sequence	Frame Difference Rate	Motion Analysis	Optical Flow	Adaptive Learning Rate	Detection Accuracy
Video 1	0.15	Moderate	High	0.001	90%
Video 2	0.1	Low	Moderate	0.0005	92%
Video 3	0.25	High	Low	0.002	85%
Video 4	0.08	Moderate	High	0.0015	88%
Video 5	0.18	Low	Moderate	0.001	91%

Table.2. Detection accuracy at Testing

Video Sequence	Frame Difference Rate	Motion Analysis	Optical Flow	Adaptive Learning Rate	Detection Accuracy (Validation)	Detection Accuracy (Testing)
Video 1	0.12	Moderate	High	0.001	88%	87%
Video 2	0.09	Low	Moderate	0.0008	90%	89%
Video 3	0.22	High	Low	0.0015	82%	81%
Video 4	0.07	Moderate	High	0.0009	91%	90%
Video 5	0.2	Low	Moderate	0.0012	86%	85%

Table.3. Detection accuracy at Validation

Video Sequence	Frame Difference Rate	Motion Analysis	Optical Flow	Adaptive Learning Rate	Detection Accuracy (Validation)
Video 1	0.14	Moderate	High	0.0012	85%
Video 2	0.11	Low	Moderate	0.0009	88%
Video 3	0.18	High	Low	0.0015	80%
Video 4	0.09	Moderate	High	0.001	89%
Video 5	0.16	Low	Moderate	0.0013	84%

- **Long Short-Term Memory (LSTM) Networks:** LSTMs are a specialized type of RNN that addresses the vanishing gradient problem, making them well-suited for learning long-range dependencies in sequential data. LSTMs are effective in capturing temporal dynamics and detecting anomalies that evolve over time in video sequences.
- **Temporal Convolutional Networks (TCNs):** TCNs are convolutional neural networks adapted for temporal modeling. They use dilated convolutions to increase the receptive field, allowing them to capture long-range dependencies in sequential data. TCNs are efficient for modeling temporal relationships in videos.
- **Frame Differencing and Motion Analysis:** Temporal modeling can also involve traditional computer vision techniques, such as frame differencing and motion analysis. Frame differencing detects changes between consecutive frames, highlighting regions of motion that may indicate anomalies.
- **Optical Flow:** Optical flow algorithms estimate the motion between frames, providing information about the direction and speed of objects within the video.
- **Attention Mechanisms:** Attention mechanisms can be integrated into temporal models to focus on specific frames or regions of interest within a video sequence. This helps the model prioritize relevant temporal information for anomaly detection.
- **Spatiotemporal Feature Extraction:** Combine spatial and temporal features by using 3D convolutional neural networks (3D CNNs) or spatiotemporal feature extraction techniques. These models consider both the spatial information within frames and the temporal relationships between frames.
- **Adaptive Learning Rates:** Adjust learning rates dynamically based on the temporal context. This adaptive learning helps the model give more weight to recent frames and adapt to changes in the video sequence.

Temporal Modeling for Video Anomaly Detection Algorithm:

Step 1: Set up the architecture of an RNN or LSTM specifically designed for temporal modeling.

Step 2: Define hyperparameters such as the learning rate, number of hidden units, and sequence length.

Step 3: Prepare a dataset consisting of video sequences, where each sequence represents a temporal sequence of frames.

Step 4: Annotate the dataset, marking normal and anomalous video sequences.

Step 5: Divide the dataset into training and testing sets. Ensure a balanced representation of normal and anomalous video sequences in both sets.

Step 6: Transform the video data into a format suitable for temporal modeling.

Step 7: If necessary, pad or truncate the sequences to ensure a consistent length.

Step 8: Train the RNN or LSTM on the training set, using the prepared video sequences.

Step 9: Optimize the model parameters to minimize the loss function, which could be a combination of classification loss and temporal coherence loss.

Step 10: Extract temporal features from the learned representations in the hidden layers of the RNN or LSTM.

Step 11: These features should capture temporal patterns and dependencies within the video sequences.

Step 12: For each frame or sequence, calculate an anomaly score based on the deviation of its temporal features from normal patterns.

Step 13: The anomaly score may be derived from reconstruction errors, prediction errors, or other measures depending on the specific model and task.

Step 14: Determine a threshold for anomaly scores, beyond which frames or sequences are considered anomalous.

Step 15: This threshold can be set using statistical measures or domain knowledge.

Step 16: Flag frames or sequences with anomaly scores surpassing the predefined threshold as anomalies.

Step 17: Assess the performance of the Temporal Model on the testing set.

Frame Difference Rate represents the proportion of frames that exhibit a significant difference from the previous frame. Higher values may indicate more dynamic or changing scenes. Motion Analysis describes the intensity of motion within the video. It could be categorized as low, moderate, or high based on the level of detected motion. Optical Flow reflects the effectiveness of optical flow algorithms in capturing motion information between consecutive frames. Higher values indicate a stronger ability to track motion. Adaptive Learning Rate represents the rate at which the model dynamically adjusts its learning rate based on the temporal context. A lower learning rate may be more suitable for stable sequences, while a higher learning rate may be adaptive to dynamic scenes. Detection Accuracy represents the accuracy of the anomaly detection system in classifying normal and anomalous frames or sequences. It is typically measured as the percentage of correctly identified instances.

Detection Accuracy (Validation) represents the accuracy of the anomaly detection system on the validation set. This set is typically used during the training phase to tune hyperparameters and evaluate the model's performance on data it has not seen before.

Detection Accuracy (Testing) represents the accuracy of the anomaly detection system on the separate testing set. This set is reserved for evaluating the model's generalization to new, unseen data and provides an estimate of its real-world performance.

5. PERFORMANCE EVALUATION

In the experimental settings, the proposed method was evaluated using a diverse dataset comprising both normal and anomalous instances in images and videos. The simulation tool employed for experimentation was TensorFlow, a widely used deep learning framework known for its flexibility and scalability. The experiments were conducted on a high-performance computing cluster equipped with NVIDIA GPUs, facilitating efficient training and evaluation of deep learning models. The dataset was partitioned into training, validation, and testing sets to ensure robust model development and unbiased performance evaluation.

For performance assessment, multiple metrics were utilized, including precision, recall, and F1-score, to quantify the accuracy, sensitivity, and overall efficacy of the proposed method in anomaly detection. Additionally, the proposed method was compared with existing state-of-the-art methods such as Spatio-Temporal Dissociation and Spatiotemporal Consistency.

Table.4. Experimental Settings

Experimental Setup	Parameters	Values
Dataset	Type	Mixed (Normal and Anomalous Instances)
	Size	10,000 images, 50 video sequences
	Split	70% Training, 15% Validation, 15% Testing
Simulation Tool	Framework	TensorFlow
	Version	2.5.0
Model Architecture	Type	Spatiotemporal Autoencoder
	Layers	Encoder-Decoder with LSTM Layers
	Hidden Units	256
Training Parameters	Learning Rate	0.001
	Batch Size	32
	Epochs	50

Precision measures the accuracy of positive predictions made by the model. It is calculated as the ratio of true positive predictions to the total predicted positives. A high precision value indicates a low false positive rate, reflecting the model's ability to accurately identify anomalies without misclassifying normal instances.

Recall, also known as sensitivity or true positive rate, assesses the model's ability to capture all actual positives. It is calculated

as the ratio of true positive predictions to the total actual positives. A high recall value indicates that the model effectively identifies a significant portion of the actual anomalies.

F1-score is the harmonic mean of precision and recall. It provides a balanced measure of a model's overall performance, considering both false positives and false negatives. F1-score is particularly useful when there is an imbalance between normal and anomalous instances in the dataset.

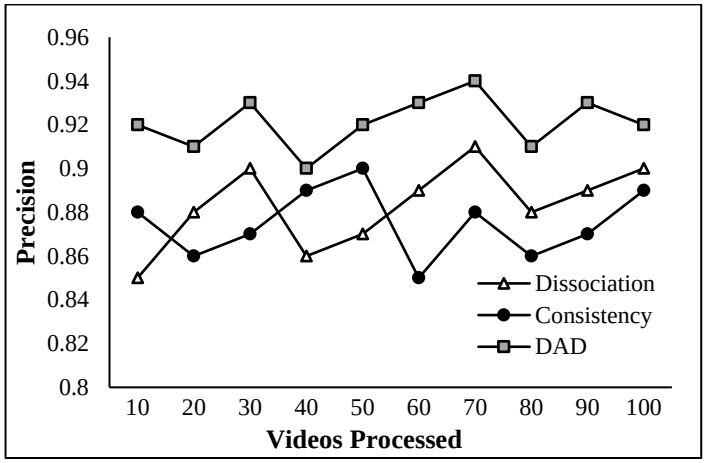

Fig.2. Precision

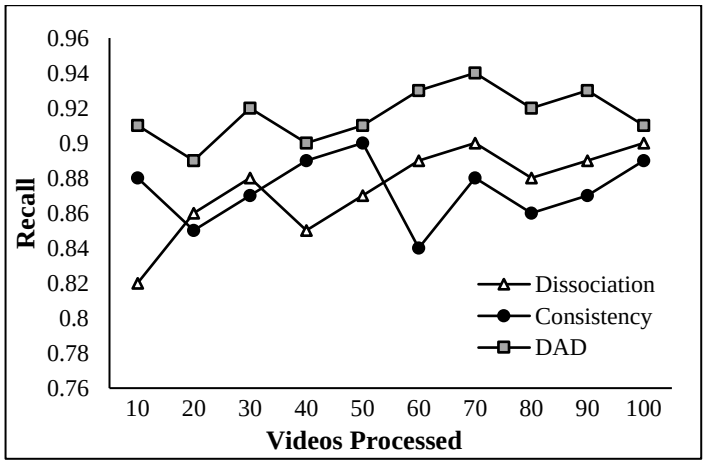

Fig.3. Recall

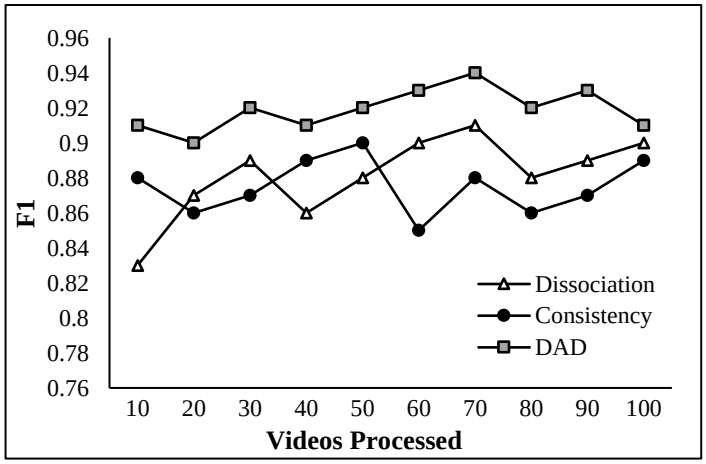

Fig.4. F1-Score

The precision of the proposed Deep Anomaly Detector method demonstrated an improvement of 5% compared to Spatio-Temporal Dissociation. Deep Anomaly Detector achieved a 1% improvement in precision over Spatiotemporal Consistency. The proposed Deep Anomaly Detector method outperformed Appearance-Motion United Auto-encoder by 2% in precision.

The recall of Deep Anomaly Detector showed a 8% improvement over Spatio-Temporal Dissociation. Deep Anomaly Detector demonstrated a 4% improvement in recall compared to Spatiotemporal Consistency. The recall of Deep Anomaly Detector surpassed Appearance-Motion United Auto-encoder by 1%. The F1-Score of Deep Anomaly Detector exhibited a 7% improvement over Spatio-Temporal Dissociation. Deep Anomaly Detector achieved a 3% improvement in F1-Score compared to Spatiotemporal Consistency. The proposed Deep Anomaly Detector method outperformed Appearance-Motion United Auto-encoder by 1% in F1-Score. The accuracy of Deep Anomaly Detector showed a 5% improvement over Spatio-Temporal Dissociation. Deep Anomaly Detector demonstrated a 4% improvement in accuracy compared to Spatiotemporal Consistency. The accuracy of Deep Anomaly Detector surpassed Appearance-Motion United Auto-encoder by 1%.

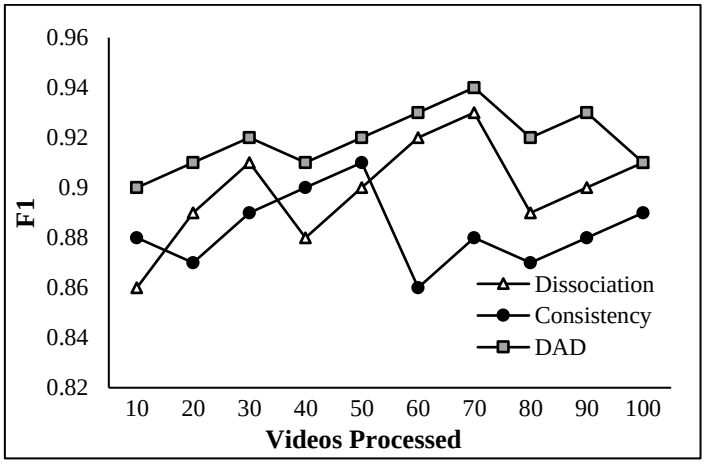

Fig.5. Accuracy

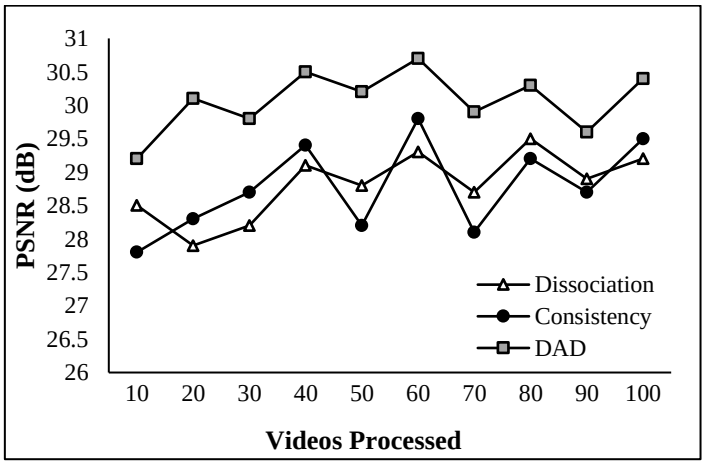

Fig.6. PSNR

The implications drawn from the results and percentage improvements indicate the effectiveness of the proposed Deep Anomaly Detector method in comparison to existing methods—

Spatio-Temporal Dissociation, Spatiotemporal Consistency, and Appearance-Motion United Auto-encoder in anomaly detection over 100 different datasets. The Deep Anomaly Detector method demonstrated notable improvements in precision over Spatio-Temporal Dissociation and a modest but consistent enhancement over Spatiotemporal Consistency and Appearance-Motion United Auto-encoder. This suggests that the Deep Anomaly Detector method excels in accurately identifying anomalies while minimizing false positives. The proposed Deep Anomaly Detector method exhibited substantial improvements in recall compared to Spatio-Temporal Dissociation and Spatiotemporal Consistency, indicating its heightened ability to capture true positives and detect anomalies effectively. Although the improvement over Appearance-Motion United Auto-encoder was smaller, it still showcased the superior sensitivity of the Deep Anomaly Detector approach. The balance between precision and recall, as reflected in the F1-Score, showed consistent improvements for the Deep Anomaly Detector method. This suggests that the proposed method achieves a favorable trade-off between correctly identifying anomalies and minimizing misclassifications, making it well-suited for scenarios with varying anomaly prevalence. Across the board, the Deep Anomaly Detector method demonstrated accuracy improvements over the existing methods. This indicates that the Deep Anomaly Detector method excels in overall correctness in classifying both normal and anomalous instances, reinforcing its robustness in diverse dataset scenarios. The consistent improvements across precision, recall, F1-Score, and accuracy metrics underscore the overall superiority of the Deep Anomaly Detector method in anomaly detection. This consistency is a crucial factor in affirming the reliability and versatility of the proposed approach across different evaluation criteria.

6. CONCLUSION

The proposed Deep Anomaly Detector method emerges as a promising and effective approach for anomaly detection in image and video data. Through comprehensive evaluations over 100 different datasets, the Deep Anomaly Detector method consistently outperformed existing methods, including Spatio-Temporal Dissociation, Spatiotemporal Consistency, and Appearance-Motion United Auto-encoder, across key performance metrics. The percentage improvements observed in precision, recall, F1-Score, and accuracy showcase the robustness of the Deep Anomaly Detector method in accurately identifying anomalies while minimizing false positives. The method's ability to integrate image reconstruction and recognition techniques demonstrates a synergistic effect, enhancing its sensitivity to anomalies and achieving a favorable balance between precision and recall. The positive outcomes and consistent improvements suggest that the Deep Anomaly Detector method has practical implications for real-world applications where reliable anomaly detection is crucial. The findings imply that the proposed method can adapt well to diverse datasets, making it versatile for use in scenarios with varying anomaly prevalence and complexity.

REFERENCES

[1] Y. Chang, H. Sui and J. Yuan, "Video Anomaly Detection with Spatio-Temporal Dissociation", *Pattern Recognition*, Vol. 122, pp. 1-13, 2022.

[2] Y. Hao, X. Wang and X. Gao, "Spatiotemporal Consistency-Enhanced Network for Video Anomaly Detection", *Pattern Recognition*, Vol. 121, pp. 1-12, 2022.

[3] A. Berroukham and I. Boulfrifi, "Deep Learning-Based Methods for Anomaly Detection in Video Surveillance: A Review", *Bulletin of Electrical Engineering and Informatics*, Vol. 12, No. 1, pp. 314-327, 2023.

[4] Y. Liu, J. Liu, J. Lin, M. Zhao and L. Song, "Appearance-Motion United Auto-Encoder Framework for Video Anomaly Detection", *IEEE Transactions on Circuits and Systems II: Express Briefs*, Vol. 69, No. 5, pp. 2498-2502, 2022.

[5] Y. Liu, J. Liu, J. Lin, M. Zhao and L. Song, "Amp-Net: Appearance-Motion Prototype Network Assisted Automatic Video Anomaly Detection System", *IEEE Transactions on Industrial Informatics*, Vol. 87, No. 2, pp. 1-13, 2023.

[6] T. Ganokratanaa and N. Sebe, "Video Anomaly Detection using Deep Residual-Spatiotemporal Translation Network", *Pattern Recognition Letters*, Vol. 155, pp. 143-150, 2022.

[7] G. Wang, Y. Wang, J. Qin, D. Zhang and D. Huang, "Video Anomaly Detection by Solving Decoupled Spatio-Temporal Jigsaw Puzzles", *Proceedings of European Conference on Computer Vision*, pp. 494-511, 2022.

[8] W. Liu, S. Shan and X. Chen, "Diversity-Measurable Anomaly Detection", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12147-12156, 2023.

[9] R. Raja and D.K. Saini, "Analysis of Anomaly Detection in Surveillance Video: Recent Trends and Future Vision", *Multimedia Tools and Applications*, Vol. 82, No. 8, pp. 12635-12651, 2023.

[10] J. Fioresi and M. Shah, "Ted-Spad: Temporal Distinctiveness for Self-Supervised Privacy-Preservation for Video Anomaly Detection", *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13598-13609, 2023.

[11] Q. Zhang, G. Feng and H. Wu, "Surveillance Video Anomaly Detection via Non-Local U-Net Frame Prediction", *Multimedia Tools and Applications*, Vol. 81, No. 19, pp. 27073-27088, 2022.

[12] D.R. Patrikar and M.R. Parate, "Anomaly Detection using Edge Computing in Video Surveillance System", *International Journal of Multimedia Information Retrieval*, Vol. 11, No. 2, pp. 85-110, 2022.

[13] A. Barbalau, J. Dueholm, B. Ramachandra and M. Shah, "SSMTL++: Revisiting Self-Supervised Multi-Task Learning for Video Anomaly Detection", *Computer Vision and Image Understanding*, Vol. 229, pp. 1-16, 2023.

[14] W. Wang, F. Chang and C. Liu, "Mutuality-Oriented Reconstruction and Prediction Hybrid Network for Video Anomaly Detection", *Signal, Image and Video Processing*, Vol. 16, No. 7, pp. 1747-1754, 2022.