International Journal Publishing
INFLUENCE: International Journal of Science Review
Volume 8, No. 3, 2026
https://influence-journal.com/index.php/influence/index

Application of the XGBoost Method with CNN Feature Extraction for AI-Generated Video Detection

Adhitya Bagus Pratikno[1], Fajri Rakhmat Umbara[2], Ridwan Ilyas[3]
[1,2,3]Universitas Jenderal Achmad Yani, Cimahi, Indonesia
Email: adhityabagus509@gmail.com

Abstract. The rapid advancement of generative artificial intelligence has substantially improved the visual quality of synthetic video, making it increasingly difficult to distinguish from authentic footage and creating new challenges for digital content verification. This study develops and evaluates an AI-generated video detection system using a hybrid approach that integrates a Convolutional Neural Network (CNN) as a feature extractor with Extreme Gradient Boosting (XGBoost) as the classifier. The research uses a 70,000-video subset of the GenVidBench-143k benchmark, selected through stratified sampling for computational efficiency. Feature extraction is performed in parallel using ResNet50, which captures high-level semantic features, and VGG16, which captures local texture detail; the resulting 2,048- and 512-dimensional vectors are concatenated into a single 2,560-dimensional representation per frame. A temporal aggregation stage transforms frame-level features into video-level features using Average Pooling and Max Pooling, and a Cost-Sensitive Learning strategy is applied through the scale_pos_weight, sample_weight, and max_delta_step parameters of XGBoost to address the marked class imbalance between real and AI-generated videos. The experimental results show that the hybrid ResNet50-VGG16 architecture combined with Average Pooling aggregation and scale_pos_weight optimization yields the best performance, achieving 92.93% accuracy, with 0.89 precision and 0.85 recall for the real-video class, and 0.94 precision and 0.96 recall for the AI-video class. A robustness analysis further shows that the model performs strongly on low-resolution video (<720p), reaching 94.00% accuracy, and maintains 88.00% accuracy on high-resolution video (>=720p); the modest decline at higher resolution reflects the increasingly seamless visual quality of modern generative models. These findings indicate that the hybrid CNN-XGBoost approach is a promising and computationally efficient strategy for synthetic video detection, although broader training-data variation is still required to reduce detection errors on high-quality video in real-world scenarios.

Keywords: *AI video Detection, Convolutional Neural Network (CNN), Extreme Gradient Boosting (XGBoost), GenVidBench, Hybrid Approach.*

A. INTRODUCTION

The rapid development of generative artificial intelligence, particularly in video content generation, has progressed at a remarkable pace in recent years. Facial-manipulation techniques (*deepfake*) and automatic text-to-video generation systems are now capable of producing synthetic video content that appears highly realistic. The resulting visuals are increasingly smooth and precise, to the point that ordinary viewers can barely distinguish them from footage captured with an actual camera (Heidari et al., 2024; Abbas & Taeihagh, 2024; Podell et al., 2023). Academic studies confirm that the sophistication of modern generative models poses a serious challenge to the verification of digital media authenticity. Whereas earlier generations of AI-generated video still exhibited easily recognizable visual artifacts, current technology can conceal such traces with far greater precision (Kuang et al., n.d.). This situation creates an urgent need for automatic detection models that are not only highly accurate but also able to adapt dynamically to the constantly evolving and increasingly sophisticated characteristics of AI-generated video.

In response to this challenge, the development of detection models today relies heavily on machine learning and deep learning because of their proven ability to learn highly complex visual patterns. Prior researchers have widely applied end-to-end deep learning methods, with

digital video forensics research relying primarily on Convolutional Neural Network (CNN) architectures to build a strong spatial feature-extraction foundation (Ismail et al., 2021). CNN-based methods have demonstrated strong performance and high accuracy when tested on laboratory datasets under clean and controlled conditions (Wang et al., 2023).

New obstacles emerge, however, when conventional CNN-based detection models are confronted with the highly dynamic conditions of the real world, where video is produced by a wide variety of AI generator models with differing technical characteristics. Variation in resolution, visual quality, and the subtlety of artifacts across modern generators frequently causes a drastic drop in detection accuracy when the model encounters new data that falls outside its training distribution (Humidan et al., 2022; Saikia et al., 2022). In addition, purely end-to-end approaches are often more difficult to train because the entire network must be optimized simultaneously, which makes the model prone to *overfitting*, a condition in which the model memorizes patterns from the training data so closely that its effectiveness declines when applied to new, unseen types of video.

To address these limitations and improve the effectiveness of detection systems, several studies have proposed a hybrid approach that structurally separates the extraction and classification stages (Mancy et al., 2025). In this scheme, the detection process is divided into two continuous stages: the first stage extracts and maps the important characteristics of an image using a CNN architecture, while the second stage performs the final classification decision using the machine learning algorithm known as Extreme Gradient Boosting (XGBoost) (Humidan et al., 2022; Khachatryan et al., 2023). Combining these two distinct methods is considered far more effective for solving the detection problem: CNN architectures are highly reliable at capturing rich and deep visual feature representations (Ismail et al., 2021), while XGBoost is known for its stability and consistency in producing classification decisions compared with relying on a single classifier alone.

Previous studies support this direction. The combination of visual feature extraction with a machine-learning classification algorithm has been shown to improve the accuracy of AI-generated content detection (Abbas & Taeihagh, 2024; Ismail et al., 2021), and CNN-XGBoost combinations tested across various generative-video datasets have proven more effective and accurate than single-classifier approaches (Khachatryan et al., 2023). This is reinforced by findings that architectures relying on a single classifier tend to experience a significant performance decline when facing video that has undergone compression or advanced manipulation (Humidan et al., 2022). The hybrid CNN-XGBoost approach has also been developed and tested on video with real-world characteristics, showing good performance under quality variation while substantially reducing the computational burden compared with a fully end-to-end deep-learning network, without sacrificing accuracy or generalization ability (Humidan et al., 2022; Khachatryan et al., 2023). Nevertheless, resolution variation and differences in visual characteristics remain a major challenge for maintaining stable performance, because manipulation artifacts are frequently erased by the compression processes applied on social-media platforms (Heidari et al., 2024; Abbas & Taeihagh, 2024).

Building on this body of work, the present study is designed to test the effectiveness of a combined detection model that merges the feature-extraction strength of two popular CNN architectures, ResNet50 for capturing global semantic structure and VGG16 for detecting local texture anomalies, with the XGBoost classification algorithm. The hybrid model is evaluated on the GenVidBench benchmark, a standard dataset that represents the characteristics of current generative-video trends (Ni et al., 2024). Beyond measuring performance on paper, this study also analyzes, in a comprehensive manner, the extent to which variation in visual quality and video resolution affects the resulting detection accuracy. Through this analysis, the study aims to contribute to the development of digital security systems, particularly by producing an AI-

generated video detection tool that is more robust, resilient, and adaptive to the future dynamics of generative technology.

B. METHOD

This study develops an AI-generated video detection system through the integration of a Convolutional Neural Network (CNN) and the XGBoost algorithm. The overall workflow is mapped systematically, beginning with the collection of the GenVidBench dataset, video preprocessing, feature extraction and feature-vector formation, classification, and finally model-performance evaluation. This methodology follows the standard machine-learning pipeline commonly used for identifying manipulated video content (Abbas & Taeihagh, 2024; Ni et al., 2024; Khachatryan et al., 2023). The complete research workflow is illustrated in Figure 1.

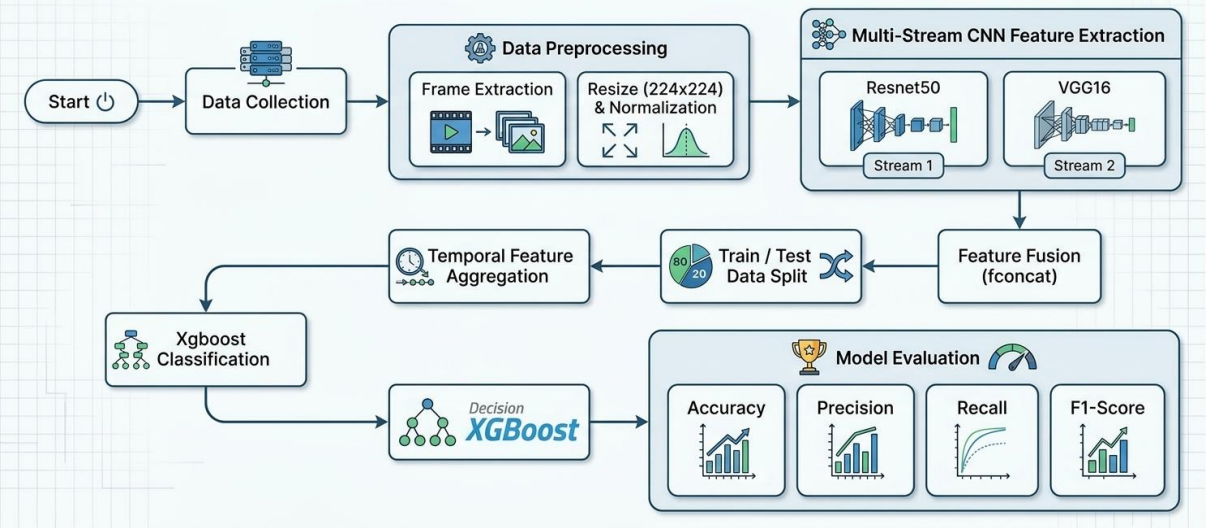

Figure 1. Overview of the Research Method

Dataset and Sampling

This study uses the GenVidBench-143k subset introduced by Ni et al. (2024), a large-scale benchmark specifically designed for evaluating AI-generated video detection systems. The subset comprises 143,400 video samples with varying resolutions and frame rates ranging from 4 to 30 FPS. Real videos originate from two real-world sources, Vript and HD-VG-130M, while the AI-generated videos are produced by eleven state-of-the-art video generator models, including Pika, VideoCrafter2, ModelScope, Text2Video-Zero, OpenSora, MuseV, SVD, Mora, CogVideo, Sora, and Kling. The composition of the dataset is summarized in Table 1.

Table 1. Composition of the GenVidBench-143k Dataset

Video Source	Class	Generative Task	Resolution	FPS	Number of Videos
Vript	Real	-	-	30	20,131
Pika	Fake	T2V & I2V	1088x560	24	13,501
VideoCrafter2	Fake	T2V & I2V	512x320	10	13,501
ModelScope	Fake	T2V	256x256	8	13,501
Text2Video-Zero	Fake	T2V	512x512	4	13,501
HD-VG-130M	Real	-	1280x720	30	13,853
MuseV	Fake	I2V	1210x576	12	13,853
SVD	Fake	I2V	1024x576	10	13,853
Mora	Fake	T2V	1024x576	10	13,853
CogVideo	Fake	T2V	480x480	4	13,853
Total	-	-	-	-	143,400

Note: T2V = Text-to-Video; I2V = Image-to-Video.

Labeling follows the official annotation provided by the dataset developer: every video receives a binary class label, *Real* for footage originating from real-world sources and *Fake* for footage produced by an AI generator model, where "1" represents a fake video and "0" represents a real video (Ni et al., 2024). Because labels are assigned according to video source, no additional manual re-labeling was required.

From the total population of 143,400 videos, this study selected a subset of 70,000 videos using a stratified random sampling technique to improve computational efficiency during CNN feature extraction and hybrid-model training while preserving the original class and generator proportions (Ismail et al., 2021; Mancy et al., 2025; Ni et al., 2024). The result of this reduction is summarized in Table 2.

Table 2. Result of Stratified Sampling from 143,400 to 70,000 Videos

Label	Generator	Proportion	143.4k	70k
0	Vript	0.1414	20,131	9,896
1	CogVideo	0.0973	13,853	6,809
0	HD-VG-130M	0.0973	13,853	6,809
1	Mora	0.0973	13,853	6,809
1	ModelScope	0.0948	13,501	6,636
1	VideoCrafter2	0.0948	13,501	6,636
1	Text2Video-Zero	0.0948	13,499	6,635
1	Pika	0.0947	13,484	6,628
1	MuseV	0.0939	13,369	6,571
1	SVD	0.0939	13,369	6,571

Based on Table 2, the stratified sampling technique successfully reduced the dataset size without altering the original percentage proportion of each class and generator. This balance is essential to keep the sample representative and to prevent model bias during training.

Preprocessing

Before feature extraction, a preprocessing stage is applied to every video to standardize the data format and improve input quality. This stage consists of three main steps: frame extraction, resizing, and pixel normalization (Ni et al., 2024).

Frame extraction. Each video is transformed into a set of two-dimensional images compatible with a CNN architecture (Heidari et al., 2024; Ismail et al., 2021). To reduce temporal redundancy and optimize computational efficiency, a *Uniform Sampling* technique based on the *Temporal Segment Network* (TSN) framework is applied: the video is divided into equal-duration segments, and one representative frame is drawn from each segment. This procedure reduces the temporal dimension of every video to a fixed length of 16 frames, standardizing the input for the subsequent temporal-aggregation stage (Humidan et al., 2022; Ni et al., 2024).

Resizing and normalization. Every extracted frame is resized to a fixed dimension of 224x224 pixels using bilinear interpolation, matching the required input size of the ResNet50 and VGG16 architectures (Ismail et al., 2021; Trivedi, 2023). Pixel values are then normalized to the 0-1 range through min-max scaling (Heidari et al., 2024; Ismail et al., 2021), ensuring that the entire dataset shares a uniform numerical scale before feature extraction and preventing the domination of large pixel values during model training.

CNN Feature Extraction

Each preprocessed frame is processed using two CNN architectures, ResNet50 and VGG16, both used purely as *feature extractors* rather than end-to-end classifiers; the final fully

connected classification layer of each network is removed so that the network only produces a numerical feature representation of the input image (Ismail et al., 2021; Trivedi, 2023).

ResNet50 applies a residual-learning mechanism that allows the network to learn much deeper feature representations without suffering from performance degradation or the *vanishing gradient* problem common in very deep conventional networks. Feature vectors are drawn from the last layer before the classification layer through *Global Average Pooling* (GAP), producing a 2,048-dimensional numerical representation per frame that captures complex visual patterns and high-level artifacts, such as structural inconsistencies and *face-warping* artifacts frequently found in AI-generated video.

VGG16 is used as a complementary feature extractor to capture more detailed local visual characteristics, such as fine texture, edges, and simple spatial patterns, that supplement the high-level features produced by ResNet50. Feature vectors are drawn from the last layer before the fully connected layer through Global Average Pooling, producing a 512-dimensional representation per frame that is particularly effective at capturing small artifacts such as inconsistent skin texture, facial transitions, or blending artifacts often present in deepfake video.

The ResNet50 (2,048-dimensional) and VGG16 (512-dimensional) feature vectors for every frame are then combined through *feature concatenation* into a single 2,560-dimensional feature vector, expressed as f_concat = [f_ResNet || f_VGG], where the operator [.||.] denotes vector concatenation along the feature dimension. This mechanism is consistent with the direct-connection approach used in the *Selective Feature Connection Mechanism* (SFCM), which merges features from multiple levels of representation prior to classification (Khachatryan et al., 2023; Trivedi, 2023). By unifying global semantic characteristics with local detail, this fusion strategy produces a richer and more discriminative representation for distinguishing real video from AI-generated video.

Temporal Aggregation, Data Splitting, and Class-Imbalance Handling

Because XGBoost requires a fixed-dimension tabular input for every sample, the feature vectors of the 16 frames belonging to each video must be reduced into a single video-level representation. This study tests two aggregation strategies separately, *Average Pooling* and *Max Pooling*. Average Pooling summarizes the general tendency of the feature distribution across all 16 frames without discarding spatial information, whereas Max Pooling selects the strongest activation value along the temporal sequence, which is designed to isolate discrete AI-generated visual artifacts that typically appear only briefly within certain frames of a sequence (Zohra et al., n.d.).

The resulting 2,560-dimensional video-level dataset is split into training and testing sets using an 80:20 ratio through *Group-Based Splitting*, in which every frame belonging to the same video is placed in the same subset. This procedure prevents *data leakage* caused by the strong temporal correlation among frames from the same video, while a *stratified sampling* scheme is additionally applied during the split to preserve the proportion of majority and minority classes as well as the representation of each AI generator across both the training and testing sets (Ni et al., 2024).

The training set exhibits a significant class imbalance, with the *Real Video* class (label 0) substantially outnumbered by the *AI Generated Video* class (label 1). To mitigate the risk that the model's loss function becomes biased toward the majority class, this study applies a *Cost-Sensitive Learning* strategy at the algorithm level rather than a data-level balancing technique, since data-level methods such as oversampling increase computational cost while undersampling risks discarding important information (Wang & Wang, 2021). Four experimental configurations are designed and compared: (1) *scale_pos_weight*, XGBoost's

built-in class-weighting parameter, calculated from the ratio of negative to positive class samples; (2) *sample_weight*, which assigns an individual weight to every training sample so that minority-class observations receive proportionally greater importance; (3) *max_delta_step*, which limits the magnitude of weight updates at every boosting iteration to stabilize optimization under extreme class imbalance; and (4) a combined configuration that applies *sample_weight* and *max_delta_step* simultaneously (Wang & Wang, 2021; Ni et al., 2024).

Classification and Evaluation

The final classification stage uses the Extreme Gradient Boosting (XGBoost) algorithm, chosen for its strong ability to handle high-dimensional feature data, its built-in regularization mechanisms that reduce overfitting, and its stable performance on binary classification tasks (Khachatryan et al., 2023). Hyperparameter optimization is carried out using *Grid Search* with cross-validation, evaluating combinations of *max depth*, *number of estimators*, *learning rate*, *subsample ratio*, and *colsample_bytree*, with the F1-score used as the primary selection criterion (Sholeh et al., 2025).

Model performance is evaluated at the video level using a *confusion matrix* consisting of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) values, from which Accuracy, Precision, Recall, and F1-Score are computed, following standard practice in AI-generated content detection research (Heidari et al., 2024; Abbas & Taeihagh, 2024). To assess robustness, the test set is additionally divided by resolution, into high-resolution (>=720p) and low-resolution (<720p) groups using video metadata obtained through FFmpeg (*ffprobe*), and the final hybrid CNN-XGBoost model is evaluated separately on each group to determine the stability of its performance under real-world quality variation (Karathanasis & Violos, 2025; Francis, 2025).

C. RESULT AND DISCUSSION

1. Preprocessing and Feature Extraction Output

Frame extraction using uniform sampling successfully reduced every video into 16 representative frames, which were subsequently resized to a uniform 224x224-pixel dimension and normalized to the 0-1 pixel-value range. This standardization ensured that the visual input format was consistent before entering the ResNet50 and VGG16 feature extractors. Each frame was then transformed into a 2,048-dimensional ResNet50 vector and a 512-dimensional VGG16 vector; the concatenation of the two produced a combined 2,560-dimensional feature vector per frame, in which indices 0-2047 contain the ResNet50 (high-level) features and indices 2048-2559 contain the VGG16 (low-level) features. This fused representation unifies global semantic characteristics with fine local detail into a single, more informative data structure for the subsequent classification stage.

2. Temporal Aggregation Experiment (Stage I)

The first experimental stage compared Average Pooling and Max Pooling as the method for reducing the 16-frame sequence into a single video-level representation, with the class-imbalance parameter fixed at scale_pos_weight = 0.3926 as a baseline so that any accuracy difference could be attributed purely to the aggregation method. The optimal hyperparameter configuration obtained through Grid Search for each aggregation scenario is presented in Table 3, and the resulting comparative performance is presented in Table 4.

Table 3. Grid-Search Hyperparameter Configuration for the Temporal Aggregation Experiment

Parameter Category	Hyperparameter	Search Space	Best Value (Average Pooling)	Best Value (Max Pooling)
Model architecture	Max Depth	[3, 5, 7]	5	5
Performance optimization	N Estimators	[100, 200, 300]	200	300
Performance optimization	Learning Rate	[0.01, 0.05, 0.1]	0.1	0.1
Performance optimization	Subsample	[0.8, 1.0]	0.8	0.8
Performance optimization	Colsample by Tree	[0.8, 1.0]	0.8	0.8
Imbalance condition	Scale Pos Weight	[0.3926]	0.3926	0.3926

Table 4. Comparative Result of the Temporal Aggregation Experiment

Aggregation Method	Target Class	Precision	Recall	F1-Score	Accuracy
Average Pooling	Real Video (0)	0.89	0.85	0.87	92.93%
Average Pooling	AI Video (1)	0.94	0.96	0.95	-
Max Pooling	Real Video (0)	0.87	0.85	0.86	92.39%
Max Pooling	AI Video (1)	0.94	0.95	0.95	-

Table 4 shows that Average Pooling achieved a slightly higher accuracy of 92.93%, compared to 92.39% for Max Pooling, a margin of 0.54%. The averaging operation proved more effective at summarizing the sequential macro-visual information continuously across the duration of the video, producing a stable global feature that made it easier for the XGBoost decision-tree algorithm to establish a consistent decision boundary. In contrast, selecting the single largest activation value in Max Pooling made the method more sensitive to extreme spatial noise, which slightly lowered precision for the minority *Real Video* class, from 0.89 to 0.87. Based on this advantage, Average Pooling was selected as the fixed temporal-aggregation method for the second experimental stage.

3. Class-Imbalance Handling Experiment (Stage II)

With Average Pooling fixed as the aggregation method, the second stage compared four class-imbalance handling strategies, evaluated using Grid Search with 3-fold cross-validation. The comparative results are presented in Table 5.

Table 5. Comparative Result of the Class-Imbalance Handling Experiment

Experimental Scenario	Target Class	Precision	Recall	F1-Score	Accuracy
Experiment 1: Scale Pos Weight	Real Video (0)	0.89	0.85	0.87	92.93%
Experiment 1: Scale Pos Weight	AI Video (1)	0.94	0.96	0.95	-
Experiment 2: Sample Weight	Real Video (0)	0.90	0.81	0.85	92.12%
Experiment 2: Sample Weight	AI Video (1)	0.93	0.97	0.95	-
Experiment 3: Max Delta Step	Real Video (0)	0.92	0.40	0.56	82.07%

Experiment 3: Max Delta Step	AI Video (1)	0.81	0.99	0.89	-
Experiment 4: Sample Weight + Max Delta Step	Real Video (0)	0.87	0.84	0.86	91.95%
Experiment 4: Sample Weight + Max Delta Step	AI Video (1)	0.94	0.95	0.94	-

Experiment 1 (Scale Pos Weight) delivered the best overall performance, with 92.93% accuracy and a stable precision-recall balance. Experiment 2 (Sample Weight) followed closely at 92.12% accuracy but was comparatively more rigid at detecting real video. Experiment 3 (Max Delta Step) showed a drastic performance decline to 82.07%, failing to reach an optimal convergence point on the complex hybrid feature space. Although the combined strategy in Experiment 4 mitigated the extreme bias observed in Experiment 3, reaching 91.95% accuracy, its effectiveness still did not surpass the single scale_pos_weight configuration.

Based on the full set of staged experiments, the final model configuration was set at the combination of hybrid CNN feature extraction (ResNet50 + VGG16), Average Pooling temporal aggregation, and XGBoost classification with scale_pos_weight optimization, achieving the highest accuracy of 92.93%, with 0.89 precision and 0.85 recall for the real-video class and 0.94 precision and 0.96 recall for the AI-video class. This configuration is superior because the 2,560-dimensional spatial feature fusion, summarized through Average Pooling, maintains continuous macro-visual representation stability throughout the video, while the scale_pos_weight value of 0.3926 effectively balances the gradient-update weight for the minority class, allowing the model to establish a strong decision boundary for detecting real video. The confusion matrix for this final model is presented in Figure 2.

Figure 2. Confusion Matrix of the Best Model (Hybrid CNN + Average Pooling + Scale Pos Weight)

4. Robustness Analysis Across Video Resolution

To assess the model's generalization capability under real-world conditions, the best-performing configuration was further tested on two resolution scenarios: high resolution (>=720p) and low resolution (<720p). The results are presented in Table 6.

Table 6. Model Performance Across Video Resolution Groups

Resolution Group	Target Class	Precision	Recall	F1-Score	Accuracy
High resolution (>=720p)	Real Video (0)	0.90	0.72	0.80	88.00%
High resolution (>=720p)	AI Video (1)	0.87	0.96	0.91	-
Low resolution (<720p)	Real Video (0)	0.74	0.94	0.83	94.00%
Low resolution (<720p)	AI Video (1)	0.99	0.95	0.97	-

On high-resolution video, the hybrid CNN-XGBoost model achieved 88.00% accuracy. The model performed strongly on the AI-video class, with a recall of 0.96 and an F1-score of 0.91; it faced more difficulty with real video, where 14 of 50 real samples were incorrectly detected as AI-generated (false positive), most likely because the sharpness of high-resolution real footage resembles the visual smoothness of modern generative output. On low-resolution video, the model reached 94.00% accuracy, with the AI-video class showing very high precision (0.99), recall (0.95), and F1-score (0.97), indicating a very low false-positive rate. This higher robustness at low resolution demonstrates the effectiveness of combining VGG16, which captures local anomaly detail, with ResNet50, which captures macro semantic features, keeping the model sensitive to subtle manipulation artifacts even after pixel compression.

The overall results are consistent with prior findings that combining visual feature extraction with a machine-learning classifier improves the accuracy of AI-generated content detection (Abbas & Taeihagh, 2024; Ismail et al., 2021), and that CNN-XGBoost combinations outperform single-classifier approaches across a variety of generative-video datasets (Khachatryan et al., 2023; Mancy et al., 2025). The advantage of Average Pooling over Max Pooling observed in this study also aligns with the general finding that mean-based temporal aggregation is more resistant to extreme spatial noise than maximum-based aggregation, which favors isolated peak activations (Zohra et al., n.d.; Bieder et al., n.d.).

The comparatively poor performance of the Max Delta Step strategy, and the fact that the combined strategy in Experiment 4 could not surpass scale_pos_weight alone, further confirms that a straightforward, ratio-based cost-sensitive weighting scheme remains a highly competitive solution for handling severe class imbalance in large-scale, high-dimensional feature spaces, consistent with the broader Imbalance-XGBoost literature (Wang & Wang, 2021). This finding suggests that increasing the complexity of an imbalance-handling strategy does not automatically translate into better performance; a single, well-calibrated parameter can outperform a more elaborate combination when the underlying feature representation is already sufficiently discriminative.

The slight accuracy decline on high-resolution video mirrors a broader challenge reported in the deepfake-detection literature: artifacts that would normally distinguish AI-generated content are progressively erased or concealed as generative models and compression pipelines improve (Humidan et al., 2022; Heidari et al., 2024; Pontorno et al., 2025). Because real footage captured at high resolution can be extremely sharp, its visual characteristics increasingly resemble the smoothness produced by modern generators, which raises the false-positive rate for the real-video class specifically. This pattern indicates that resolution-aware evaluation, rather than a single aggregate accuracy figure, is essential for understanding how a detection model will behave once deployed on heterogeneous, real-world social-media content (Karathanasis & Violos, 2025; Francis, 2025).

D. CONCLUSION

This study developed and evaluated a hybrid AI-generated video detection system that combines CNN-based feature extraction (ResNet50 and VGG16) with XGBoost classification, implemented through a complete pipeline of preprocessing, spatial feature extraction, temporal feature aggregation, and class-imbalance handling. The staged experiments show that Average

Pooling is a more effective temporal-aggregation strategy than Max Pooling for this task, and that XGBoost's scale pos weight parameter is the most effective class-imbalance handling strategy among the four configurations tested. The resulting best model, which combines hybrid CNN feature extraction, Average Pooling aggregation, and scale pos weight optimization, achieved 92.93% accuracy on the GenVidBench-143k test set, with a balanced precision-recall profile for both the real-video and AI-video classes. A robustness analysis further confirms that the model generalizes well across resolution conditions, reaching 94.00% accuracy on low-resolution video and maintaining 88.00% accuracy on high-resolution video, with the modest decline at higher resolution attributable to the increasingly seamless visual quality of modern generative models rather than a fundamental weakness of the hybrid architecture.

These findings confirm that the hybrid CNN-XGBoost approach offers a computationally efficient and reasonably accurate strategy for synthetic-video detection, while also revealing an important limitation: because training and evaluation were focused on the characteristics of the GenVidBench dataset, the model's generalization to AI-generated video produced by generators with substantially different artifact distributions remains untested. Future research is encouraged to explore alternative feature-extraction methods, such as Optical Flow or Vision Transformer (ViT) representations, to capture temporal information more effectively; to evaluate the hybrid CNN-XGBoost pipeline on larger and more heterogeneous datasets with more extreme resolution and compression variation; and to compare CNN-based feature extraction against text-representation models or other deep-learning architectures, such as LSTM or diffusion-based models, in order to identify more efficient and accurate alternatives for detecting AI-generated video in real-world scenarios.

REFERENCES

Abbas, F., & Taeihagh, A. (2024). Unmasking deepfakes: A systematic review of deepfake detection and generation techniques using artificial intelligence. *Expert Systems with Applications, 252*(PB), 124260. https://doi.org/10.1016/j.eswa.2024.124260

Bieder, F., Sandkühler, R., & Cattin, P. C. (n.d.). *Comparison of methods generalizing max- and average-pooling*.

Chen, H., & Wang, X. (2024). VideoCrafter2: Overcoming data limitations for high-quality video diffusion models. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7310-7320.

Ding, G., Sener, F., & Yao, A. (2023). Temporal action segmentation: An analysis of modern techniques. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *46*(2), 1011-1030.

Fernandes, Y. A., & Fatma, Y. (2025). Metode deep learning dalam teknologi deepfake: Systematic literature review. *JATI (Jurnal Mahasiswa Teknik Informatika)*, *9*(2), 3403-3410.

Francis, N. (2025). *Deepfake detection and defense: An analysis of techniques and robustness*. Graduate Thesis and Dissertation post-2024.

Heidari, A., Jafari Navimipour, N., Dag, H., & Unal, M. (2024). Deepfake detection using deep learning methods: A systematic and comprehensive review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *14*(2), e1520. https://doi.org/10.1002/widm.1520

Hong, W., Ding, M., Zheng, W., Liu, X., & Tang, J. (2022). Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*.

Huang, Z., He, Y., Yu, J., Zhang, F., Si, C., Jiang, Y., ... & Liu, Z. (2024, June). Vbench: Comprehensive benchmark suite for video generative models. In *2024 IEEE/CVF*

Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 21807-21818). IEEE.

Humidan, A. S., Abdullah, L. N., & Halin, A. A. (2022, December). Detection of compressed deepfake video drawbacks and technical developments. In *2022 5th International conference on signal processing and information security (ICSPIS)* (pp. 11-16). IEEE.

Ismail, A., Elpeltagy, M., S. Zaki, M., & Eldahshan, K. (2021). A new deep learning-based methodology for video deepfake detection using XGBoost. *Sensors*, *21*(16), 5413.

Jolliffe, I. T., & Cadima, J. (2016). Principal component analysis: A review and recent developments. *Philosophical Transactions of the Royal Society A, 374*(2065), 20150202. https://doi.org/10.1098/rsta.2015.0202

Karathanasis, A., Violos, J., & Kompatsiaris, I. (2025). A comparative analysis of compression and transfer learning techniques in deepfake detection models. *Mathematics*, *13*(5), 887.

Khachatryan, L., Movsisyan, A., Tadevosyan, V., Henschel, R., Wang, Z., Navasardyan, S., & Shi, H. (2023, October). Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)* (pp. 15908-15918). IEEE.

Khalid, M., Raza, A., Younas, F., Rustam, F., Villar, M. G., Ashraf, I., & Akhtar, A. (2024). Novel sentiment majority voting classifier and transfer learning-based feature engineering for sentiment analysis of deepfake tweets. *IEEE Access*, *12*, 67117-67129.

Kuang, L., Wang, Y., Hang, T., Chen, B., & Zhao, G. (2022). A dual-branch neural network for DeepFake video detection by detecting spatial and temporal inconsistencies. *Multimedia Tools and Applications*, *81*(29), 42591-42606.

Li, J., Zhang, C., Zhu, W., & Ren, Y. (2025). A comprehensive survey of image generation models based on deep learning. *Annals of Data Science*, *12*(1), 141-170. https://doi.org/10.1007/s40745-024-00544-1

Mancy, H., Elpeltagy, M., Eldahshan, K., & Ismail, A. (2025). Hybrid-Optimized Model for Deepfake Detection. *International Journal of Advanced Computer Science & Applications*, *16*(4), 148.

Ni, Z. L., Qiangyu, Y. A. N., Yuan, T., Huang, M., Hu, H., Chen, X., & Wang, Y. (2025). *Genvidbench: A challenging benchmark for detecting ai-generated video*.

Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., ... & Rombach, R. (2024, May). Sdxl: Improving latent diffusion models for high-resolution image synthesis. In *International Conference on Learning Representations* (Vol. 2024, pp. 1862-1874).

Pontorno, O., Guarnera, L., & Battiato, S. (2025). DeepFeatureX-SN: Generalization of deepfake detection via contrastive learning. *Multimedia Tools and Applications*, *84*(39), 47721-47740.

Saikia, P., Dholaria, D., Yadav, P., Patel, V., & Roy, M. (2022, July). A hybrid CNN-LSTM model for video deepfake detection by leveraging optical flow features. In *2022 international joint conference on neural networks (IJCNN)* (pp. 1-7). IEEE.

Sholeh, M., Lestari, U., & Andayati, D. (2025). Hyperparameter Optimization Using Grid Search and Random Search to Improve the Performance of Prediction Models with Decision Trees. *Jurnal Riset Multidisiplin dan Inovasi Teknologi*, *3*(03), 453-464.

Trivedi, A. K., Mahajan, T., Maheshwari, T., Mehta, R., & Tiwari, S. (2025). Leveraging feature fusion ensemble of VGG16 and ResNet-50 for automated potato leaf abnormality detection in precision agriculture: AK Trivedi et al. *Soft Computing*, *29*(4), 2263-2277.

Wang, C., Deng, C., & Wang, S. (2020). Imbalance-XGBoost: leveraging weighted and focal losses for binary label-imbalanced classification with XGBoost. *Pattern recognition letters*, *136*, 190-197.

Wang, J., Yuan, H., Chen, D., Zhang, Y., Wang, X., & Zhang, S. (2023). Modelscope text-to-video technical report. *arXiv preprint arXiv:2308.06571*.

Zohra, F., Zhao, C., Liu, S., & Ghanem, B. (2025, June). Effectiveness of max-pooling for fine-tuning clip on videos. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (pp. 3282-3291). IEEE.