International Journal of Computer Applications (0975 - 8887)
Volume 187 - No.147, October 2026

Interpretable Texture, Frequency and Structural Features for Explainable Fake Image and Deepfake Detection

Shailendra Singh Kathait
Co-Founder and Chief Data Scientist
Valiance Solutions
Noida, India

Ashish Kumar
Principal Data Scientist
Valiance Solutions
Noida, India

Dheeraj Kumar Prajapati
Intern Data Scientist
Valiance Solutions
Noida, India

Riya Singh
Intern Data Scientist
Valiance Solutions
Noida, India

ABSTRACT

Generative models have significantly increased the realism of synthetic images, which makes reliable fake image detection a challenging problem. Deep learning detectors achieve high accuracy, but they are difficult to interpret and often generalize poorly to unseen generators. This paper presents a set of handcrafted, interpretable features that capture intrinsic differences between real and synthetic images from three complementary perspectives: texture complexity, frequency-domain repetition, and geometric consistency. Texture richness is quantified by Local Binary Pattern (LBP) entropy and local density, repetitive upsampling artifacts are measured through Fourier spectrum analysis, and structural consistency is measured through edge-based line statistics and structure-tensor coherence. Each feature is defined formally and analyzed in terms of computational cost, expected behavior, and known confounding factors, and an evaluation protocol covering image, video, cross-generator, and post-processing scenarios is specified. The features are deployed in a detection platform, where they provide feature-level explanations for the prediction of an image classifier, and a video pipeline aggregates frame-level decisions into a video-level verdict. The descriptors are computationally efficient, model-agnostic, and suitable for classical machine learning as well as hybrid detection frameworks.

General Terms: Image Forensics, Pattern Recognition.

Keywords: Fake Image Detection, Deepfake Detection, Explainable Detection, Texture Analysis, Frequency Artifacts, Local Binary Patterns (LBP), Structure Tensor

1Introduction

Recent advances in deep generative models, including Generative Adversarial Networks (GANs) and diffusion-based models [12,13], have enabled the synthesis of highly realistic images. These models offer significant benefits in applications such as data augmentation, content creation, and design, but they also introduce serious risks related to misinformation, identity fraud, and malicious image manipulation [28].

Most existing fake image detection methods rely on supervised deep learning models trained on specific datasets, or on large language models that identify visual anomalies. Although these approaches achieve high accuracy under controlled settings, they often generalize poorly to unseen generative techniques [22, 29] and lack interpretability because of their black-box nature.

In contrast, handcrafted features derived from image statistics and physical imaging properties are transparent, require fewer computational resources. Such features capture intrinsic artifacts introduced during image generation and are less dependent on the architecture or training data of a specific generative model.

This paper focuses on extracting low-level and mid-level features that reveal subtle inconsistencies in synthetic images. Three characteristics are targeted:

(1) texture homogeneity caused by over-smoothing;
(2) periodic artifacts introduced by convolutional upsampling operations;
(3) geometric irregularities resulting from imperfect structural modeling.

By integrating spatial, frequency-domain, and geometric analyses, an interpretable feature representation for fake image detection is constructed. Section 6 describes how the same features are applied to the detection of fake videos.

The main contributions of this work are:

—a set of interpretable features that view an image from texture, frequency, and geometric perspectives, each formally defined;
—an adaptation of structure-tensor density and coherence analysis to image authenticity;
—an analysis of each feature covering parameter settings, computational complexity, expected behavior, and confounding factors;
—an evaluation protocol spanning several datasets and scenarios, supported by a reproducible evaluation script;
—a deployment of the features in a platform that returns feature-level explanations for images and videos.

The remainder of the paper is organized as follows. Section 2 reviews related work, Section 3 defines the proposed features,

Section 4 analyzes them, Section 5 specifies the evaluation protocol, Section 6 presents the deployment and qualitative results, and Section 7 concludes the paper.

2Related Work

Deep learning approaches have dominated fake image detection in recent years. Convolutional Neural Networks (CNNs) trained on paired real–fake datasets have shown impressive performance [29]. XceptionNet [3] and EfficientNet architectures have been widely adopted for deepfake detection [27]. However, these methods often struggle to generalize to unseen generative models or to training data from different distributions [4].

Recent work has explored Vision Transformers (ViTs) and CLIP-based models for synthetic image detection [5, 22], which achieve better cross-generator generalization. Despite their success, these approaches remain largely black-box systems, which makes it difficult to understand which visual cues drive their decisions. Post-hoc explanation methods such as Grad-CAM [26] and LIME [24] show where a network looked, but not which image property is unusual.

Early image forensics research focused on detecting compression artifacts, resampling traces [23], and inconsistencies in sensor noise patterns [10]. Texture-based descriptors such as Local Binary Patterns (LBP) have been widely adopted for face spoofing detection [19] and image tampering detection because of their robustness to illumination variations and their computational efficiency.

Frequency-domain methods have been effective in identifying periodic artifacts introduced by upsampling and convolution operations in GAN-based architectures [7, 9, 30]. The origin of such artifacts in transposed convolutions is documented in [20], and several studies report that synthetic images exhibit anomalous peaks in the Fourier spectrum that are typically absent in real images [8].

Geometric consistency analysis, including edge continuity, line coherence, and structural alignment, has also been explored for detecting image manipulation. Synthetic images often fail to preserve long-range structural dependencies, which results in fragmented or inconsistent edges. Orientation coherence derived from the structure tensor [1] offers a compact, well-established measure of such structure.

Unlike data-intensive deep learning approaches, the proposed method does not rely on large-scale training datasets, which makes it suitable for practical and interpretable fake image detection. Benchmarks such as FaceForensics++ [25], Celeb-DF [18], and GenImage [31] provide the data on which such features can be evaluated, as outlined in Section 5.

3Proposed Methodology

Given an RGB image $I \in \mathbb{R}^{H\times W\times 3}$, a set of handcrafted features is extracted to capture texture characteristics, frequency-domain regularities, and structural consistency. Grayscale conversion is applied where necessary, and all features are computed at the native resolution of the input image. Most algorithms are implemented with the OpenCV library [14]. Fig. 1 shows the overall feature-extraction pipeline.

3.1Texture Complexity

Motivation Real images generally exhibit complex and diverse texture patterns because of natural variations in illumination, material properties, and sensor noise. Fake images are hypothesized to display smoother textures and reduced randomness.

Method The grayscale image is processed with the uniform Local Binary Pattern (LBP) [21] operator with radius $r = 1$ and $P = 8$ sampling points, which yields ten distinct codes. A normalized histogram h of LBP codes is computed over the entire image, and texture richness is summarized as the Shannon entropy (natural logarithm) of the histogram:

$$\mathscr{T}_{\mathrm{LBP}} = -\sum_{i=1}^{10} h_i \log(h_i). \tag{1}$$

Expected behavior Higher entropy indicates greater texture diversity. Over-smoothed synthetic regions are expected to shift the entropy, and the direction of the shift on a given data set is determined empirically (Section 5).

3.2Repeating Pattern Detection

Motivation Generative models often introduce subtle periodic artifacts through convolutional filters and upsampling layers [20]. Such artifacts are difficult to detect in the spatial domain but become prominent in the frequency domain.

Method The grayscale image is transformed with the 2D Fast Fourier Transform (FFT). The magnitude spectrum is converted to a logarithmic scale, $M = 20\log|F|$ (natural logarithm), and centered. To suppress low-frequency components, a circular mask of radius 10 pixels is applied around the spectrum center. The scalar feature is the 85th percentile of the positive values of the masked spectrum:

$$\mathscr{F}_{\mathrm{rep}} = \mathrm{Percentile}_{85}(M_{\mathrm{masked}}). \tag{2}$$

Expected behavior Higher values indicate stronger periodic frequency peaks, which are associated with images generated by deep models.

3.3Structural Line Consistency

Motivation Natural images tend to preserve coherent geometric structures such as edges and lines. Fake images often exhibit broken or inconsistent line formations because of limited global structural understanding.

Method Canny edge detection [2] is applied to the grayscale image, followed by the Probabilistic Hough Transform [6] for line detection. The Canny hysteresis thresholds are 100 and 150. The Hough transform uses a distance resolution of 1 pixel, an angular resolution of one degree, an accumulator threshold of 50, a minimum line length of 30 pixels, and a maximum gap of 5 pixels. The length ℓ_i of each of the N detected segments is computed, and two features are extracted. The average line length is

$$\mathscr{L}_{\mathrm{avg}} = \frac{1}{N}\sum_{i=1}^{N} \ell_i, \tag{3}$$

and the ratio of short to long lines is

$$\mathscr{L}_{\mathrm{ratio}} = \frac{|\{\ell_i \le \mathscr{L}_{\mathrm{avg}}\}|}{|\{\ell_i > \mathscr{L}_{\mathrm{avg}}\}|}. \tag{4}$$

When no segment is detected, or no segment is longer than the average, the corresponding values are treated as undefined.

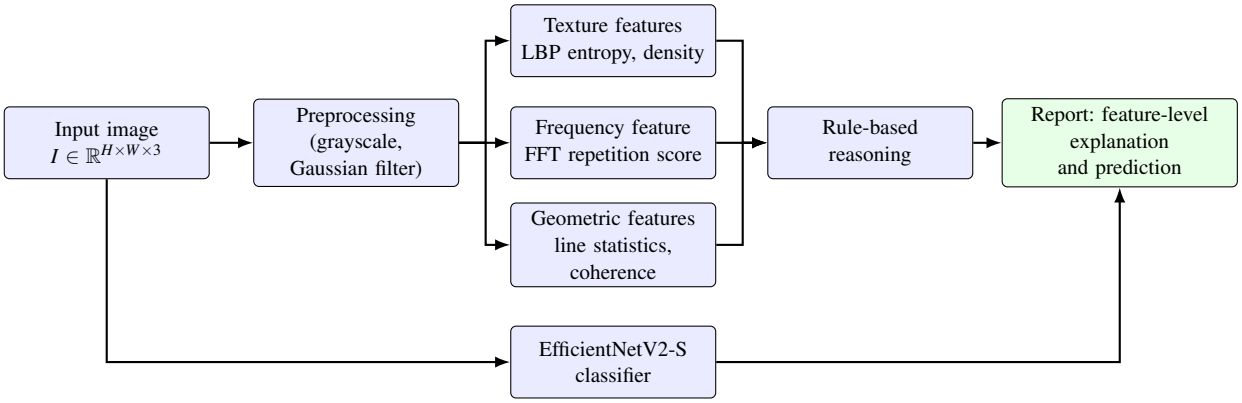


Fig. 1. Overview of the proposed pipeline. The handcrafted features drive the explanation, while an EfficientNetV2-S classifier produces the authenticity prediction.

Expected behavior Fake images are expected to produce shorter, fragmented line segments, which results in higher short-to-long ratios.

3.4Structure-Tensor Texture and Coherence Analysis

Many fake images appear visually realistic but fail to preserve the long-range structural consistency and directional coherence of real-world images. Natural images exhibit locally consistent orientations and coherent edge structures because of physical scene constraints, whereas synthetic images frequently show fragmented orientations, irregular coherence, and abnormal local density patterns. To capture these discrepancies, a structure-tensor-based analysis [1] that quantifies local density and directional coherence is employed. The implementation is adapted from the open-source PyTextureAnalysis framework [17] and repurposed for image authenticity analysis.

3.4.1Preprocessing and normalization The input image is converted to an 8-bit grayscale image $G \in [0,255]^{H\times W}$. To reduce high-frequency noise while preserving structure, Gaussian filtering [11] is applied:

$$G_f = \mathscr{G}_\sigma(G), \tag{5}$$

where σ controls the scale of local smoothing.

3.4.2Local density estimation

Motivation Fake images often exhibit unnatural intensity distributions, such as excessive uniformity or irregular clustering in textured regions. Measuring local density helps to identify such deviations.

Method Local density is computed from the original grayscale image by convolution with a normalized square kernel of size k:

$$D(x,y) = \frac{1}{k^2} \sum_{(i,j)\in\Omega_k} G(x+i, y+j). \tag{6}$$

The density map is normalized to $[0,1]$, and a pixel is counted as high-density if $D(x,y) \geq \tau_d$. The percentage of high-density pixels is

$$\mathscr{D}_{\%} = \frac{|\{(x,y) : D(x,y) \geq \tau_d\}|}{H \times W} \times 100. \tag{7}$$

Expected behavior Abnormally low or high density percentages indicate synthetic texture anomalies. Real images typically show moderate, spatially varying density, whereas fake images may display excessive smoothness or irregular clustering.

3.4.3Coherence analysis

Motivation Coherence measures how strongly a local region exhibits a dominant orientation. Real images preserve coherent structure because of physical scene constraints, whereas fake images may show disrupted or inconsistent local orientations.

Method First, image gradients are computed with Sobel operators, $G_x = \partial G_f/\partial x$ and $G_y = \partial G_f/\partial y$. Second, the 2D structure tensor is formed as

$$\mathbf{J} = \begin{bmatrix} \langle G_x^2\rangle_\sigma & \langle G_x G_y\rangle_\sigma \\ \langle G_x G_y\rangle_\sigma & \langle G_y^2\rangle_\sigma \end{bmatrix}, \tag{8}$$

where $\langle\cdot\rangle_\sigma$ denotes Gaussian smoothing with standard deviation σ. Since $\mathbf{J}$ is real, symmetric, and positive semi-definite, its eigenvalues are real and non-negative, ordered as $\lambda_1 \leq \lambda_2$. Coherence at each pixel is

$$\mathscr{C}(x,y) = \frac{\lambda_2 - \lambda_1}{\lambda_1 + \lambda_2 + \varepsilon}, \tag{9}$$

where ε is a small constant for numerical stability. Pixels below an intensity threshold are excluded to suppress noise-dominated regions. The valid coherence values are aggregated into a normalized two-bin histogram, and the percentage of pixels in the low- and high-coherence regimes is

$$\mathscr{C}_{\text{low}}, \mathscr{C}_{\text{high}} = \frac{100}{N} \sum_{(x,y)} \mathbf{1}\big(\mathscr{C}(x,y) \in \mathscr{B}_{\text{low/high}}\big), \tag{10}$$

where N is the number of valid pixels and $\mathbf{1}(\cdot)$ is the indicator function.

Expected behavior High coherence corresponds to well-defined edges or linear structures, which are common in real images. Low coherence reflects isotropic or disordered regions, which are expected to be more frequent in fake images.

3.5Feature Summary

Table 1 summarizes the extracted features.

Table 1. . Summary of the extracted features

Category	Feature	Symbol	Description
Texture	LBP entropy	$\mathscr{T}_{\text{LBP}}$	Measures texture diversity
Frequency	FFT repetition score	$\mathscr{F}_{\text{rep}}$	Captures periodic upsampling artifacts
Geometry	Average line length and short-to-long ratio	$\mathscr{L}_{\text{avg}}, \mathscr{L}_{\text{ratio}}$	Measures continuity of edges and lines
Texture	Density percentage	$\mathscr{D}_{\%}$	Captures abnormal intensity clustering
Texture	Coherence histogram	$\mathscr{C}_{\text{low}}, \mathscr{C}_{\text{high}}$	Measures consistency of local orientation

Table 2. . Fixed settings of the feature extractors and the platform

Component	Setting
LBP	Uniform, $r = 1$, $P = 8$ (10 codes)
FFT score	Mask radius 10 px, 85th percentile of positive values
Canny	Hysteresis thresholds 100 and 150
Hough	1 px, 1°, accumulator 50, min. length 30 px, max. gap 5 px
Classifier	EfficientNetV2-S, 224×224 input, threshold 0.5
Video sampling	Every 10th frame (24 FPS $\rightarrow$ 2.4 FPS)

4Analysis of the Proposed Features

This section documents the configuration of the feature extractors, their computational cost, and the factors that are expected to influence each feature, so that the behavior of the descriptors can be understood independently of any single data set.

4.1Configuration

Table 2 lists the fixed settings used by the implementation. The kernel size k, the density threshold τ_d, the smoothing scale σ, and the stability constant ε of the structure-tensor module are configuration parameters of that module.

4.2Computational Complexity

For an image with $H \times W$ pixels, the LBP histogram requires $O(HWP)$ operations, which is $O(HW)$ for the fixed $P = 8$. The FFT requires $O(HW \log(HW))$ operations and dominates the cost of the pipeline. Canny edge detection, the Sobel gradients, and the Gaussian smoothing of the structure tensor are $O(HW)$ for fixed kernel sizes, and the box filter used for local density is $O(HW)$ as well. The probabilistic Hough transform depends on the number of edge pixels, which is at most $O(HW)$, and on the number of angular bins. The memory requirement is $O(HW)$. None of the extractors needs a GPU or a training stage, which supports deployment in resource-constrained environments.

4.3Expected Behavior and Confounding Factors

Because the features are computed from low-level image statistics, they respond to any process that alters those statistics, not only to image generation. Table 3 summarizes, for each feature, the cue that is exploited and the factors that can confound it. In particular, JPEG compression introduces an 8×8 block grid that produces periodic spectral peaks, and resizing or resampling introduces correlations of the kind exploited in resampling forensics [23]. These effects can mimic, mask, or amplify the upsampling traces of a generator. Features that rely on absolute pixel thresholds, such as the minimum Hough line length, also depend on the image resolution. For this reason, the evaluation protocol in Section 5 includes post-processing and resolution scenarios.

4.4Comparison with Learned Detectors

Table 4 contrasts the proposed descriptors with learned detectors along the criteria that motivate this work. The generalization entry for the handcrafted features is a design expectation rather than a result, and it is the subject of the protocol in Section 5.

5Evaluation Protocol and Scenarios

A single dataset cannot show whether handcrafted features generalize, so the evaluation is organized around several datasets and scenarios, summarized in Table 5. Real and fake face images and videos from FaceForensics++ [25] and Celeb-DF [18] cover face manipulation, and GenImage [31] supplies images from several GAN and diffusion generators for the cross-generator scenario.

5.1Metrics

For each feature, the mean and standard deviation are reported separately for real and fake images, together with the single-feature area under the ROC curve (AUC), computed with the Mann–Whitney statistic with ties counted as 0.5. The AUC is defined as the probability that the feature value of a fake image exceeds that of a real image. An AUC above 0.5 therefore indicates that fake images tend to have higher values, and an AUC below 0.5 indicates lower values, so the direction of each feature is established empirically rather than assumed. For the rule-based reasoning module, the share of images that trigger each rule is reported for real and fake images, and, when the rules are used as a detector, accuracy, fake recall, and real specificity are reported. In S3, the video-level accuracy of each aggregation strategy is reported.

5.2Reproducibility

The evaluation is implemented as a script that applies the same feature functions as the platform to labeled folders of real and fake images and writes the per-image feature table, a summary, and ready-to-use LaTeX tables. The script is available from the authors upon request.

6Deployment and Qualitative Results

The proposed framework is deployed in the *Valiance Deepfake Detection Platform*. The platform allows users to upload images or videos and returns an automated authenticity assessment

Table 3. . Exploited cue and known confounding factors of each feature

Feature	Exploited cue	Known confounding factors
$\mathscr{T}_{\text{LBP}}$	Loss of fine texture diversity in over-smoothed regions	Blur, denoising, beauty filters, and shallow depth of field in real photographs; sensor noise level; compression
$\mathscr{F}_{\text{rep}}$	Periodic peaks from upsampling layers	JPEG block grid, resampling, screen capture patterns, watermarks, and image size
$\mathscr{L}_{\text{avg}}, \mathscr{L}_{\text{ratio}}$	Fragmented or broken edges and lines	Scene content (portraits versus architecture), absolute length thresholds, defocus blur, and image resolution
$\mathscr{D}_{\%}$	Abnormal intensity clustering	Exposure, lighting conditions, and background brightness
$\mathscr{C}_{\text{low}}, \mathscr{C}_{\text{high}}$	Disrupted local orientation	Hair and fabric patterns, noise, cropping, and the intensity threshold used to exclude pixels

Table 4. . Handcrafted features versus learned detectors

Criterion	Proposed features	Learned detectors
Explanation	Explicit feature values	Post-hoc saliency maps
Training data	Not needed to compute features	Large labeled sets
Unseen generators	Expected to rely on generic artifacts; to be validated	Often degraded [4, 29]
Hardware	CPU, FFT-dominated	Typically GPU
Post-processing	Sensitive (Table 3)	Varies by model

accompanied by feature-level explanations that highlight potential indicators of synthetic generation. For images, the authenticity prediction is produced by an EfficientNetV2-S classifier (input resized to 224×224, sigmoid output thresholded at 0.5), and the handcrafted features generate the explanation that accompanies it. Fig. 2 shows the user interface, which supports image and video inputs. Upon submission, the content is processed by the feature-extraction pipeline, and the analysis is summarized in a structured report.

6.1Image-Level Analysis

Fig. 3 presents representative outputs generated by the platform for AI-generated images. In addition to the final authenticity prediction, the platform reports interpretable cues derived from texture entropy, frequency-domain irregularities, and structural inconsistencies. These explanations indicate the visual artifacts that contributed to the decision, such as reduced texture diversity, abnormal frequency peaks, and fragmented geometric structures.
In Fig. 3(a), the image exhibits excessive smoothness, and unnatural skin and hair texture is reported as the dominant indicator of synthetic generation.
Although the image in Fig. 3(b) appears visually realistic, the platform reports broken and incoherent lines as the reason for its classification as fake. These two examples are illustrative; they show the form of the explanation rather than measuring detection performance, which is the purpose of the protocol in Section 5. They also show that the handcrafted features can provide transparent, human-interpretable explanations alongside the prediction, in contrast to purely deep learning-based detectors, which typically offer limited insight into the causes of a fake classification.

6.2Video Processing Pipeline

To extend the framework to temporal data, the video processing pipeline in Fig. 4 is used. To balance computational efficiency with detection reliability, video inputs are temporally downsampled by extracting every tenth frame. For a source rate of 24 frames per second (FPS), this yields an effective processing rate of 2.4 FPS. Each sampled frame undergoes facial localization and tracking based on the architectures described in [15] and [16], respectively. The identified regions of interest (ROIs) are cropped and processed by the feature-extraction module. Finally, frame-level predictions are aggregated by majority voting or weighted averaging to determine a video-level verdict, which ensures that transient errors in isolated frames do not compromise the overall assessment.
This frame-wise strategy is designed to detect manipulated facial content while remaining scalable for resource-constrained deployments, and its modular design allows the interpretable features to be integrated into larger forensic and monitoring systems.

7Conclusion

This paper presented an interpretable, handcrafted feature-based framework for diagnosing images and identifying plausible causes of synthetic manipulation. By jointly examining texture complexity, frequency-domain regularities, and geometric consistency, the proposed features capture complementary aspects of image realism. Texture entropy targets over-smoothing, frequency-domain analysis targets periodic artifacts introduced by generative pipelines, and geometric and structure-tensor features target structural inconsistencies. The paper also analyzed the computational cost, configuration, and confounding factors of each feature, and defined an evaluation protocol over several datasets and scenarios.
Unlike other deep learning-based approaches that primarily provide binary predictions, the proposed method computes explicit feature values that offer insight into the visual artifacts behind a fake classification. The framework is model-agnostic, computationally efficient, and transparent in its explanations, which makes it suitable for resource-constrained environments and forensic scenarios where explainability is critical. The extracted features can also be integrated with classical or hybrid machine learning classifiers. The code is available from the authors upon reasonable request at info@valiancesolutions.com.

Table 5. . Evaluation scenarios

ID	Scenario	Data	Purpose
S1	In-distribution faces	Face frames from FaceForensics++ and Celeb-DF	Separability of each feature on face regions of interest
S2	Cross-generator images	GenImage subsets, one per generator, GAN and diffusion families	Per-generator behavior and consistency of the direction of each feature
S3	Video level	Videos from S1 sampled every 10th frame	Effect of majority voting versus weighted averaging
S4	Post-processing	S1 and S2 images after JPEG compression, resizing, blur, and noise	Robustness of each feature to the confounders in Table 3
S5	Resolution control	S1 and S2 images resized to a fixed size before extraction	Dependence on native image resolution

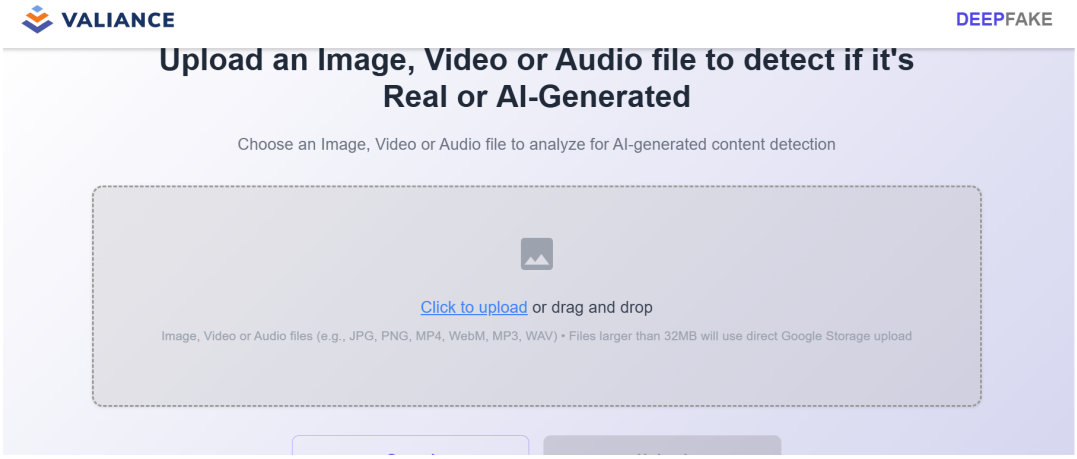


Fig. 2. User interface of the deepfake detection platform.

7.1Future Directions

Future work will execute the evaluation protocol at scale across emerging generative architectures and real-world datasets, and will extend the feature set to higher-level physical inconsistencies such as unrealistic lighting, shadow inconsistencies, reflection and refraction anomalies, and violations of fine-scale physical details. Furthermore, the framework will be extended to enable the spatial localization of these inconsistencies within images. Adaptive feature fusion strategies will also be explored to improve robustness, interpretability, and generalization.

8References

[1] Josef Bigün and Gösta H. Granlund. Optimal orientation detection of linear symmetry. In *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, pages 433–438, 1987.

[2] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, 1986.

[3] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 1251–1258, 2017.

[4] Riccardo Corvi, Davide Cozzolino, Giada Zingarini, Giovanni Poggi, Koki Nagano, and Luisa Verdoliva. On the detection of synthetic images generated by diffusion models. In *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2023.

[5] Davide Cozzolino, Giovanni Poggi, Riccardo Corvi, Matthias Nießner, and Luisa Verdoliva. Raising the bar of ai-generated image detection with clip. arXiv preprint arXiv:2312.00195, 2023.

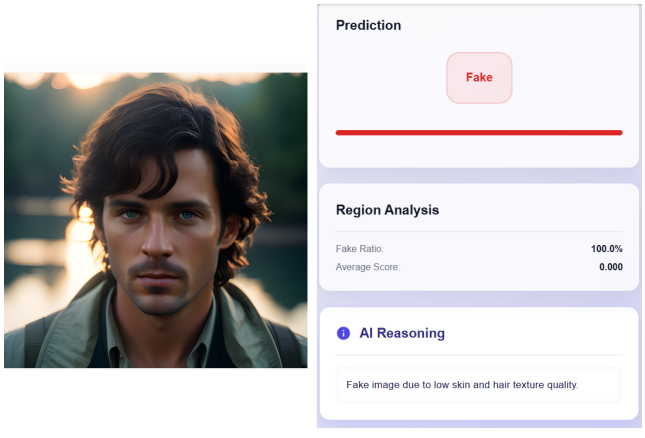
[6] Richard O. Duda and Peter E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(1):11–15, 1972.

[7] Ricard Durall, Margret Keuper, and Janis Keuper. Watch your up-convolution: CNN based generative deep neural networks are failing to reproduce spectral distributions. In *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 7890–7899, 2020.

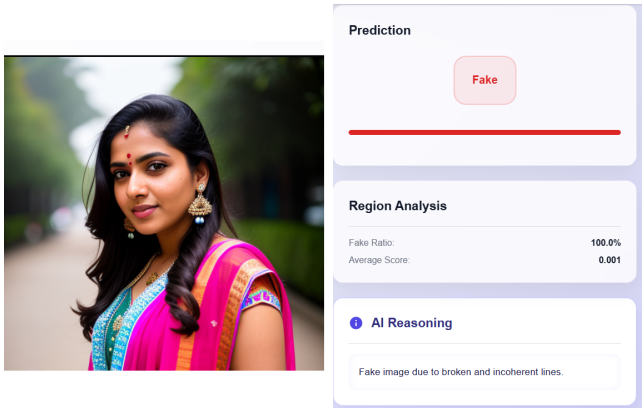
[8] Tarik Dzanic, Karan Shah, and Freddie Witherden. Fourier spectrum discrepancies in deep network generated images. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[9] Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. In *Proc. Int. Conf. Machine Learning (ICML)*, pages 3247–3258, 2020.

[10] Jessica Fridrich and Jan Kodovský. Rich models for steganalysis of digital images. *IEEE Trans. Information Forensics and Security*, 7(3):868–882, 2012.



(a)



(b)

Fig. 3. Detection results and feature-based reasoning for two AI-generated images: (a) first image, (b) second image.

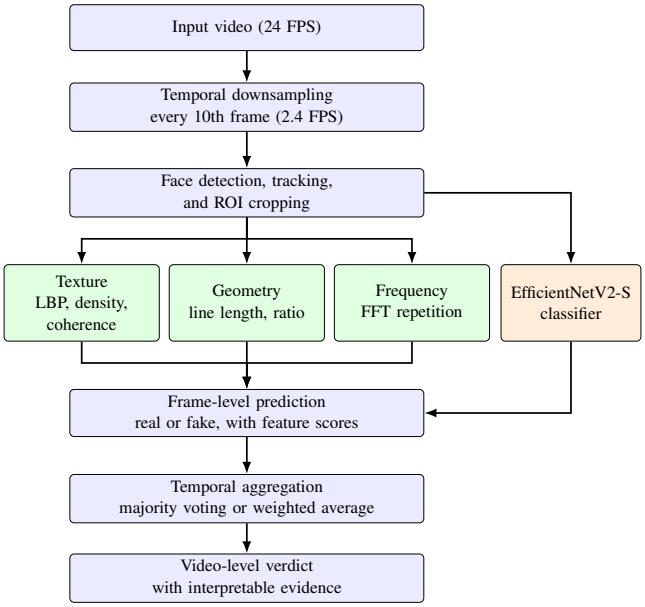


Fig. 4. Video processing pipeline.

[11] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Pearson, 4th edition, 2018.

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2014.

[13] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[14] Itseez. *Open Source Computer Vision Library*, 2015.

[15] Shailendra Singh Kathait and Ashish Kumar. Deep learning-based person tracking using facial recognition: A smart approach to security and civic monitoring. *International Journal of Computer Applications*, 187(7):1–5, May 2025.

[16] Shailendra Singh Kathait, Ashish Kumar, Samay Sawal, Ram Patidar, and Khushi Agrawal. Deep learning-based person tracking: A smart approach to security and civic monitoring. *International Journal of Computer Applications*, 187(4):1–4, May 2025.

[17] Ajinkya Kulkarni. PyTextureAnalysis, January 2023.

[18] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-DF: A large-scale challenging dataset for DeepFake forensics. In *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2020.

[19] Jukka Määttä, Abdenour Hadid, and Matti Pietikäinen. Face spoofing detection from single images using micro-texture analysis. In *Proc. Int. Joint Conf. Biometrics (IJCB)*, 2011.

[20] Augustus Odena, Vincent Dumoulin, and Chris Olah. Deconvolution and checkerboard artifacts. *Distill*, 2016.

[21] Timo Ojala, Matti Pietikäinen, and Topi Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.

[22] Utkarsh Ojha, Yuheng Li, and Yong Jae Lee. Towards universal fake image detectors that generalize across generative models. In *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 24480–24489, 2023.

[23] Alin C. Popescu and Hany Farid. Exposing digital forgeries by detecting traces of resampling. *IEEE Trans. Signal Processing*, 53(2):758–767, 2005.

[24] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In *Proc. ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2016.

[25] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. FaceForensics++: Learning to detect manipulated facial images. In *Proc. IEEE/CVF Int. Conf. Computer Vision (ICCV)*, 2019.

[26] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, pages 618–626, 2017.

[27] Maryam Taeb and Hongmei Chi. Comparison of deepfake detection techniques through deep learning. *Journal of Cybersecurity and Privacy*, 2(1):89–106, 2022.

[28] Luisa Verdoliva. Media forensics and DeepFakes: An overview. *IEEE J. Selected Topics in Signal Processing*, 14(5):910–932, 2020.

[29] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A. Efros. CNN-generated images are surprisingly easy to spot... for now. In *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 8695–8704, 2020.

[30] Xu Zhang, Svebor Karaman, and Shih-Fu Chang. Detecting and simulating artifacts in GAN fake images. In *Proc. IEEE Int. Workshop on Information Forensics and Security (WIFS)*, 2019.

[31] Mingjian Zhu, Hanting Chen, Qiangyu Yan, Xudong Huang, Guanyu Lin, Wei Li, Zhijun Tu, Hailin Hu, Jie Hu, and Yunhe Wang. GenImage: A million-scale benchmark for detecting AI-generated image. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2023.