International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(10s) 2026 pp. 39-54
https://doi.org/10.70917/ijcisim-2026-3565
Article

A Unified Deep Learning Framework for Face Forgery Detection in Varying Image Qualities Using TNVS-Net

M. Ramya[1], J. Suguna[2]

[1]Department of Computer Science, Vellalar College for Women (Autonomous), Erode-12.
Email: ramyasivanathan@gmail.com
[2] Department of Computer Science, Vellalar College for Women (Autonomous), Erode-12.
Email: sugunajravi@gmail.com

Abstract: Face forgery is the manipulation or synthetic generation of facial images or videos to falsely represent someone's identity, expressions, or actions. Deep Learning (DL) methods have been widely used for both creating and detecting such forgeries. Various DL methods have been developed for face forgery detection and perform well on high-quality images and videos. However, real-world images are often compressed by social media platforms degrading visual quality, blurring fine details and introducing noise and artifacts that hinder detection quality. To address this, Multi-Scale Dual-Branch Network (MDCF-Net) is used which combines multi-scale Red Green Blue (RGB) and frequency domain features to detect manipulated faces. These features are processed through a Spatial Pyramid Pooling (SPP) layer to produce a fixed-length representation followed by a softmax classifier for prediction. But, MDCF-Net is mainly designed for detecting forgeries in highly compressed facial images. However, it has not yet attained the best possible performance in detecting forgeries on datasets that are not compressed or have modest compression. Hence, in this paper, Nexus Relation Network (NRN) branch along with RGB domain branch and a frequency domain branch of MDCF-Net is proposed. This proposed model termed as Three-Node Variable-Scale Network (TNVS-Net) has been developed for detecting highly compressed, low compressed or uncompressed facial images. The developed model is tested with three different datasets and found that it achieves higher performance than the existing classical face forgery prediction models.
Keywords: Face Forgery, Deep Learning, Multi-Scale RGB, Frequency Filter, Nexus Relation Network

1. INTRODUCTION

Forgery is the act of creating, altering, or imitating documents, signatures, images, works of art, or objects with the intent to deceive others [1]. It is a serious crime often committed for financial gain, identity theft or misrepresentation. There are several types of forgeries based on what is being falsified. Signature forgery involves faking someone's handwritten signature; document forgery includes altering legal records; art forgery refers to copying original artworks; currency forgery involves printing fake money and identity forgery steals personal details to commit fraud [2]. A growing modern threat is face forgery which involves digitally manipulating or synthesising facial images or videos to impersonate individuals or spread misinformation [3]. Techniques like face swapping, facial re-enactment and synthetic face generation enable the creation of highly realistic but fake content [4].

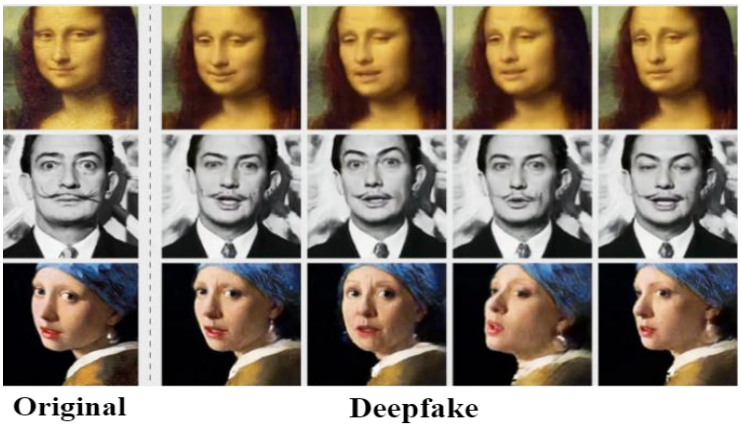


Figure 1 Original vs. Deep fake

The growing prevalence of face forgery is largely driven by rapid advancements in Deep Learning (DL), specific subfield of Artificial Intelligence (AI) that enables machines to automatically understand and recognize intricate patterns from large sets of visual data [5]. Deep fake technology combines deep learning and fake content to generate realistic images or videos that are difficult for humans or machines to detect.Deep fake technology utilize models like Generative Adversarial Networks (GANs), that allows for the production of digitally altered media with lifelike realism by altering faces, expressions, and speech [6]. Initially used for entertainment, it is now misused for political disinformation, identity fraud, and extortion [7]. Tools such as Faceswap, Zao, and DeepFaceLab have made deep fake creation accessible, raising serious concerns about media trust and security [8]. Public figures like Barack Obama and Joe Biden have been targeted, highlighting the threat [9, 10].To combat this, organizations like DARPA, Facebook, and Google have launched initiatives to develop detection techniques [11]. DL techniques like Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs) were trained to detect subtle inconsistencies in synthetic media [12–14]. However, many existing methods perform well on test datasets but struggle with cross-dataset generalization and degraded, compressed media often found on social platforms [15].ViTs offer a promising solution by leveraging self-attention mechanisms to analyze inter-patch relationships and detect manipulated regions [16]. Their multi-scale feature extraction helps identify forgery traces even in low-quality content, where traditional RGB-based approaches become less effective [17, 18].

Forgery traces in the RGB domain are often subtle but can become more apparent when analyzed in the frequency domain, where they may manifest as compression anomalies or abnormal frequency distributions [19]. Unlike genuine regions, manipulated areas typically exhibit irregular spectral patterns. Deep learning models improve detection by using feature fusion and attention mechanisms to capture multi-level information from images. However, many conventional methods rely solely on RGB features and overlook frequency-based cues. By integrating both RGB and frequency domain information, models can better capture fine-grained forgery artifacts, resulting in improved identification of deep fake.

For instance, MDCF-Net [20] was proposed to leverage distinctive changes in the frequency domain caused by many face forgery techniques. This model combines features from both the frequency and RGB domains to detect manipulated faces. This model effectively captures cross-domain fake cues at multiple scales, particularly in compacted facial images. Its RGB branch employs transformers to derive fine-grained, textural elements derived from the source RGB images at multiple scales. Simultaneously, the frequency domain branch analyzes poor quality films for artifacts by identifying and utilizing global spectral attributes. A multi-head attention-based fusion module then integrates spatial and frequency features in a complementary manner. To produce a fixed-length feature depiction, a Spatial Pyramid Pooling (SPP) layer is applied, followed by a dropout-based classifier that performs binary classification to decide the image as original or duplicate. However, MDCF-Net is primarily designed for detecting forgeries in highly compressed facial images and has yet to achieve optimal performance on datasets with little or moderate compression.

2. LITERATURE SURVEY

Despite growing advancements in deepfake detection, current methods face key challenges such as poor generalization to unseen forgeries, high computational costs, and limited performance on uncompressed or complex data. Many rely on static optimization or small datasets, reducing accuracy and adaptability. This survey reviews recent approaches in light of these gaps, emphasizing the need for more efficient and generalizable solutions.

Tran et al. [21] created a Meta Deepfake Detection (MDD) method for detecting unseen deepfake domains without requiring model updates. Multitask Cascaded Convolutional Network (MTCNN) was utilized

for face extraction.Pair-Attention Loss, softmax loss and Average-Center Alignment Loss were used in training and prediction part of the MDD model.The MDD method employs meta-weight learning to optimize origin to destination knowledge transfer and domain representation via meta-optimization.But, this model relies solely on meta-optimization during training and does not update during testing on new domains.

Sun et al. [22] created Fake Trajectory Detection Network (FTDN) for face forgery prediction. A virtual-anchor-based technique was initially developed to extract the trajectory of face area displacement. The collected displacement data was input into a Dual Stream Spatial-Temporal Graph Attention (DSSTGA) and Gated Recurrent Unit. The DSSTGA model uses temporal and spatial features in trajectory series to learn adaptive graph attention weights, combining GRU and dense encoders to extract and fuse spatiotemporal embeddings for deepfake classification. But, the additional dense layers might lead to high computational and inference time.

Alnaim et al. [23] introduced the Deepfake Face Mask Dataset (DFFMD) for deepfake detection using Inception-ResNet-v2 model. Initially, the pre-processing was performed to extract detailed image information. Inception-ResNet-v2 was used to extract features for enhancing the likelihood of identifying the images as authentic or duplicate. Residual connection in the multiple-sized convolutional filters ofInception ResNet block eliminates feature deterioration and training time. But, lower accuracy was resulted due to limited training epochs and potentially insufficient or less complex dataset scale.

Ramadhani et al. [24] suggested adeep fakerecognitionmodelusing the Improved Video Vision Transformer (IViViT) structure.The input video undergoes pre-processing to extract 25 facial landmarks per frame forming a sequences. Then, tubelets were created and features were extracted using Depthwise Separable Convolution (DSC) with Convolution Based Attention Module (CBAM). These attributes were compressed, positionally determined and analyzed through Spatio-temporal processing in ViVT's multihead attention blocks followed by classification using a sigmoid function. But, using only landmark areas limits positional encoding, potentially reducing spatial coherence for key detection cues.

Liu et al. [25] developed a deep forgery detection method that integrates deep neural networks (DNNs) with fine-grained artifact featuresto identify real and fake images under various disturbances. The dual-branch classification strategy includes forgery detection branch using Convolutional Neural Network (CNN) and residual detection branch that upsamples images and extract residuals under transformation. The upsampled residual maps of actual and falseimages utilized to train the CNN in the forgery detection module. But, this method was inherently limited in handling complex forgery data lowering the accuracy.

Zhou et al. [26] developed Intra-Inter Network- Forged Trace Masking Module (IIN-FTMM) was suggested to detect the face forgeries in videos. An intra-module was used to supervised learning to learn about various kinds of face forgeries and extract unique features, whereas the inter-module uses self-supervised learning to learn about common features. In addition, a Forged Trace Masking Module (FTMM) was used for contrastive learning in face forgery detection. But, the significant variation in forgery features across different datasets makes it difficult to extract shared features reducing the model's generalization capability.

Wang et al. [27]created a model for facial forgery recognition that utilized Adversarial Training and Feature Disentanglement (ATFD). Adversarial training generated forged samples by targeting specific facial regions to mimic forgery effects enhancing data diversity. Feature disentanglement was guided by mutual information loss with an added adversarial loss to enhance the focus on forgery-specific features. This dual approach significantly improved the overall efficiency of the modelin identifying various types of face forgeries.But, the model needs to focus on unseen forgery types and data distributions lower the models ability to perform on diverse dataset.

Khormali et al. [28] developed a Self-Supervised Graph Transformer (SSGT) technique for deep fakerecognition. This model involves a feature extractor based on ViT structure was pre-trained through self-supervised contrastive learning technique, a graph convolution network united with a transformer discriminator that transform the extracted feature as a graph feature and a graph transformer relevancy map that identify potentially manipulated facial regions. But, this model struggles to capture the semantic relationships between facial regions fades the relevant information.

Soudy et al. [29] suggested Deepfake detection using convolutional ViT and CNN. This model has three stages: preprocessing, detection, and prediction. Pre-processing involves Frame extraction, facial identification, face position, face trimming, eye trimming, and nose trim. During the detection phase, CNNs were employed to identify features on the eyes and nose, and a CNN in conjunction with a ViT was employed to identify faces. To arrive at individual forecasts during the prediction phase, the majority vote method involved combining the outcomes of the three models that were applied to the three distinct features. But, this technique may struggle to detect deepfakes involving facial areas beyond the eyes, nose, and full face.

Siddiqui et al. [30] presented a CrossViT with a DenseNet-based convolutional feature extractor model for deep fake prediction. By combining local and global feature analysis, this method improves the detection of tiny modifications. By examining the information collected from the hybrid architecture, the model is able to forecast and categorize video frames as authentic or not. However, this model results in high computational complexity and fails to capture deep complex artifacts.

While recent methods offer promising directions, most still struggle with adaptability, computational efficiency, and detection under diverse conditions. These limitations highlight the need for models that learn consistent features across varied data. A three-branch approach shows potential in addressing these gaps, offering a more robust path forward in deepfake detection.

3. PROPOSED METHODOLOGY

This division depicts the pipeline of the suggested TNVS-Net model for detecting highly compressed, low compressed or uncompressed face forgeries in Figure 2. The network architecture primarily comprises the A multi-scale RGB branch employs Transformers to extract fine-texture features from real RGB images, while a frequency filter domain branch in LSF captures artifacts in videos with poor quality by obtaining global spectral attributes. NRN branch directs the network in acquiring knowledge about the consistent features of distributions. All these branches assists to overcome the challenges of detecting face forgeries in highly compressed, low compressed or uncompressed facial images.

3.1 Shallow and Deep Features

This module employs the ImageNet pre-trained EfficientNet-B4 as a backbone for facial feature extraction, offering an ideal equilibrium between precision and computational effectiveness. EfficientNet-B4 consists of seven layers, where the second layer focuses on extracting fine-grained texture features, and the sixth layer captures global semantic features. Pre-processed and cropped real face images, denoted as $X \in \mathbb{R}^{H\times W\times C}$, are passed through this backbone to generate shallow texture feature maps $T \in \mathbb{R}^{(H/4)\times(W/4)\times C}$, here H and W denote the image dimensions and C denotes the channel count. These extracted attributes are fed into the model's three key components: the Multi-scale RGB Branch, the Frequency Filter Branch, and the Neural Relation Network (NRN) Branch for further processing for forgery detection.

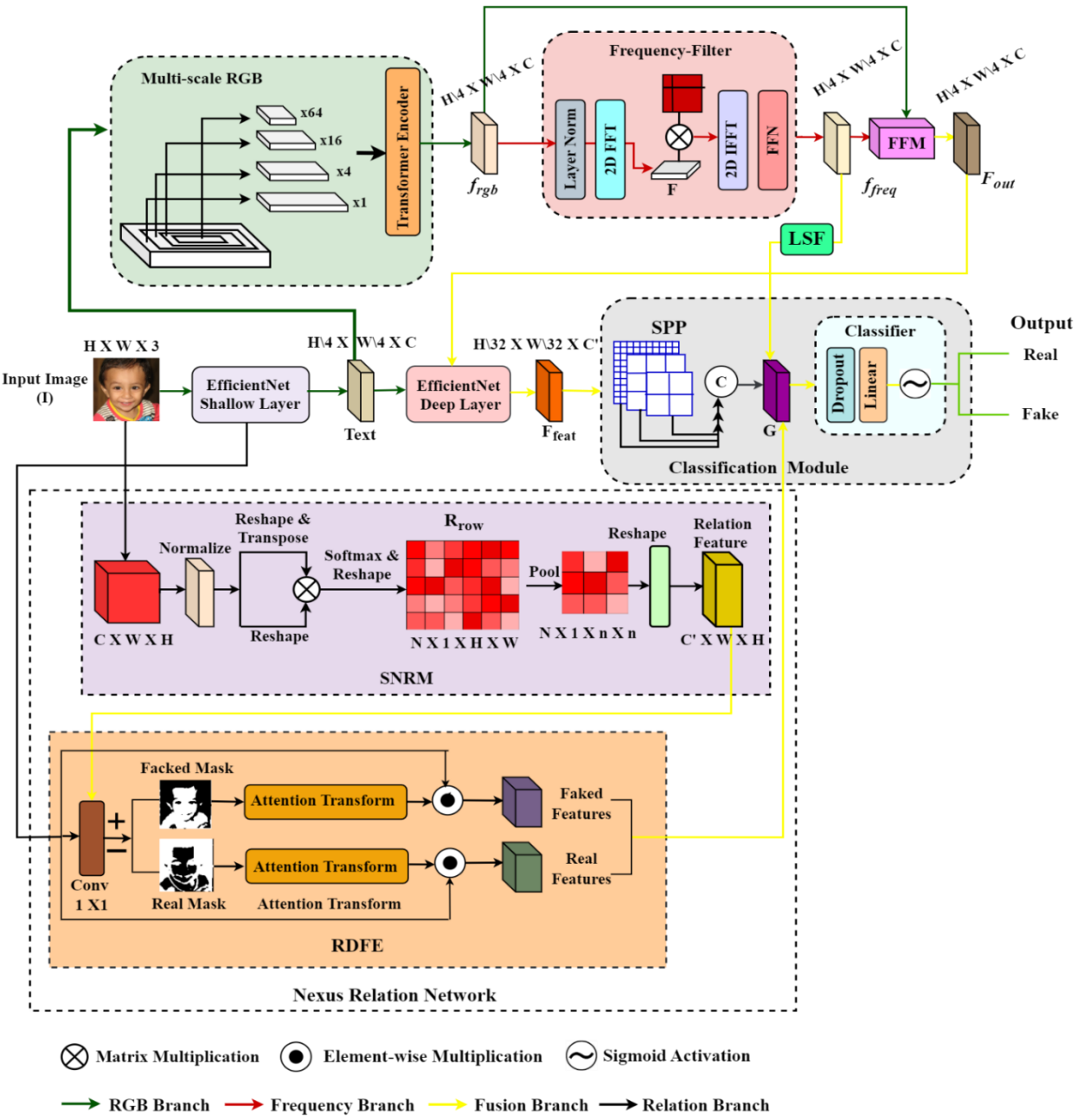


Figure 2 Schematic Representation of the proposed model

3.2 Multi-scale RGB Branch

The multi-scale RGB module's depth concept utilizes the transformer's self-attention mechanism. The input to this module is the texture feature map $T \in \mathbb{R}^{(H/4)\times(W/4)\times C}$, extracted from the backbone network EfficientNet-B4. To capture multi-scale contextual information, the input feature map is divided into patches at four different scales: the original scale, anddownscaled by factors of $1/2$, $1/4$ and $1/8$.Each patch size is denoted as $C_h = r_h \times r_h \times C$, where r_h represents the spatial resolution of the patch at scale h. These patches are then independently passed through self-attention layers with each scale treated as a separate attention head. By using the matrix multiplication and softmax operation, the proximity p_x^h between various image patched is computed by Eq. (1)

$$p_x^h = softmax\left(\frac{q_x^h (k_x^h)^T}{C_h}\right) \tag{1}$$

where, q_x^h, k_x^h are the query and key embeddingsfor the x^{th} image patch at scale h. The resemblance values of relevant image patches are weighted and summed to determine the result of the query image patch P_x^h as

$$P_x^h = \sum_{x=1}^{N} p_x^h v_x^h \tag{2}$$

In Eq. (2), v_x^h denotes the value embedding for the x^{th} image patch at scale h. This process is repeated across multiple patch scales, enabling the model to capture features at diverse resolutions. The outputs from each scale are reshaped into feature tensors and concatenated to form a unified representation of four parts. This reshaped result is transformed into the original image shape, resulting in the output as $f_{rgb} \in \mathbb{R}^{H/4 \times W/4 \times C}$.

The above module is designed to strengthen the feature extraction process by capturing detailed and context-aware features at multiple scales, which is essential to the overall network's capability to detect highly compressed, low compressed or uncompressed facial images accurately.

3.3 Frequency-filter Domain

The spatial features determined after transformer encoding is converted into the frequency domain to extract frequency information. The frequency filter module [31] initially executed by applying a 2D Fast Fourier Transform (2D-FFT) to the texture map f_{rgb} resulting in its complex frequency domain (F) representation

$$F = \mathcal{F}\left[f_{rgb}\right] \in \mathbb{R}^{H/4 \times W/4 \times C} \tag{3}$$

where, $\mathcal{F}\left[f_{rgb}\right]$ represents the 2D-FFT operation. F is element-wise multiplication by a learnable complex weight matrix $W \in \mathbb{R}^{H/4 \times W/4 \times C}$, which enables adaptive frequency selection,

$$\hat{F} = W \odot F \tag{4}$$

where, $\odot$ represents the Hadamard product. Since, W shares the similar dimensions as F, it functions as a global filter where F represents arbitrary frequency-domain filter. Finally, an inverse FFT is applied to the filtered spectrum $\hat{F}$, non-linearly transforming it back into the spatial domain to generate the final frequent feature f_{freq}, as in Eq. (5)

$$f_{freq} \leftarrow \mathcal{F}^{-1}\left[\hat{F}\right] \tag{5}$$

where, $\mathcal{F}^{-1}\left[\hat{F}\right]$ denotes the inverse FFT. This process allows the model to capture long-range relationships in the frequency domain while preserving fine-grained local details from the spatial domain, thereby improving the downstream detection and classification performance.

Although frequency feature representations are compatible with CNNs, they must first be projected back into the spatial domain, which limits the direct utilization of spectral information. Extracting CNN features directly from the frequency domain often proves ineffective for forgery detection. To address this, Learnable Frequency Selection (LFS) is introduced. LFS captures discriminative frequency information while maintaining the shift-invariance and local consistency characteristic of natural RGB images. These enhanced frequency features are then passed through EfficientNet-B4 to extract high-level patterns indicative of forgery. The LFS operates on frequency features of dimension $H/4 \times W/4 \times C$, providing a localized focus to detect subtle and abnormal frequency distributions. As illustrated in Figure 3, LFS uses a sliding window (SW) mechanism, where h denotes the adaptive statistical aggregation across each spatial grid.The LFS calculation are computed within defined frequency bands enables a more compact statistical representation while producing a smoother distribution that is less affected by outliers.

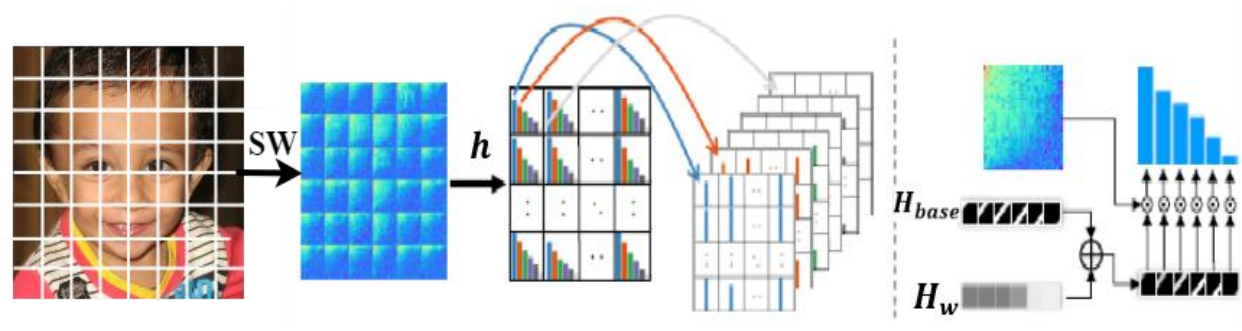


Figure 3 LFS Module

Specifically, for each windowq ∈ a, after applying the FFT, the local statistics are computed by summing the amplitudes within each frequency band. For each band, the statistic is defined by Eq. (6)

$$u_x = log_{10}\|\mathcal{D}(q)\odot [H_{base}^x + \varphi(H_w^x)]\|_1, x = \{1, \dots, T\} \quad (6)$$

Importantly, the log_{10}function is employed to balance the magnitude across each frequency band. The frequency spectrum is divided equally into *T*bands, arranged sequentially from low to high frequency. Here,H_{base}^x and H_w^xrepresent the base and learnable filters for the x^{th} band, respectively. The local frequency statistics *u* for each window *q* are then reshaped into a $1 \times 1 \times$ Tvector. Collecting these vectors from all windows forms a matrix with a down sampled spatial size relative to the input image, and the number of channels equal to *T*. This matrix serves as the input to subsequent convolutional layers.

The LFS module processes a *299×299×3*input image by sliding *10×10* windows with a stride of 2. Each window's frequency spectrum is divided into 6 bands, resulting in a compact output matrix of size *149×149×6*. This matrix captures localized frequency features, enabling effective detection of face forgeries. By using LFS in the frequency-filter branch, the model gains a more compact, localized, and outlier-resistant representation of the differences between fake and real content in the frequency domain, thereby improving forgery detection performance.

3.4 Feature Fusion Module

The FFM [20] is utilized to receive the input feature maps $f_{rgb} \in \mathbb{R}^{H/4 \times W/4 \times C}$ and $f_{freq} \in \mathbb{R}^{H/4 \times W/4 \times C}$. This FFM is a fusion module with multi-head attention employing a cross modal attention mechanism which effectively integrate features from both RGB and frequency domains enhancing their complementarity. Specifically, three 1×1 convolutional operations are applied to map the spatial features to queries (Q), and the frequency features to keys (K) and values (V). These projections are then split into *N* separate heads, resulting in *N* distinct sets of Q_x, K_x, and V_x,*where x = 1, 2, …, N*. Each head independently performs attention computation as defined by Eq.(7)

$$E_x = softmax\left(\frac{Q_x(f_{rgb})K_x(f_{freq})^T}{\sqrt{d_{k_x}}}\right)V_x(f_{freq}) \quad (7)$$

where, $Q_x(f_{rgb})$, $K_x(f_{freq})$ and $V_x(f_{freq})$ are utilized to produce queries, keys and values for the multi-head attention mechanism. This strategy is set to use 3 heads, meaning the feature inputs are split into three parts, and attention is calculated separately for each part. So, the model may emphasis on different details or patterns in the RGB and frequency features simultaneously.The outputs of all heads are concatenated along the channel dimension, passed through a convolutional layer to match the desired output channels, and then merged to the original RGB input through a residual connection, yielding the final fused feature output F_{out}

$$M = Concatenate(M_1, M_1, .., M_1) \quad (8)$$

$$F_{out} = f_{rgb} + Conv(M) \quad (9)$$

In Eq. (9), $Conv(M)$ transforms the fused features into the desired number of output channels, utilizing a 1×1 convolutional layer, a batch normalization layer and Leaky ReLU(Rectified Linear Unit) for processing.Similar to the original Transformer, the residual connection improves information flow and prevents gradient vanishing during deep network training. Multi-head attention enables each head to capture diverse features, allowing the model to better understand face image content. Then, the multi-scale RGB and frequency will be stacked repetitively. At last, $F_{out} = \mathbb{R}^{(H/4)\times(W/4)\times C}$ are inputted to the deep convolutional layers of the backbone network resulting in additional extracted deep features $F_{feat} = \mathbb{R}^{(H/32)\times(W/32)\times C}$.

The FFM module effectively fuses RGB and frequency features using multi-head attention, enhancing complementary details and enabling deeper, more robust feature representations for accurate face forgery detection.

3.5 Nexus Relation Network Branch

To leverage spatial relations between highly compressed, low compressed or uncompressed facial images, dualdifferent-scaled relation modules are designed i.e., the SNRM and the RDFE.

3.5.1 Spatial Neighbor Relation Module (SNRM): The Spatial Neighbor Relation Module (SNRM) is designed to capture spatial relationships between features of every pair of pixels before performing pixel-wise classification. Given an input feature map $F_x \in \mathbb{R}^{C\times H\times W}$, the features of individual pixel are first normalized. This is then reshaped to$Z \in \mathbb{R}^{C\times N}$, where, $N = H \times W$ represents the total number for pixels in the feature maps.

To model the spatial relationships, a spatial relation map $SR \in \mathbb{R}^{N\times N}$ is computed by multiplying Z and Z^T, where T represents the transpose.This results in a matrix that encodes the similarity between each pair of pixels. The resulting spatial relation map is then normalized along its second dimension using the softmax function, as described by Eq. (10)

$$SR_{x,y} = \frac{\exp(Z_x^T \times Z_y)}{\sum_{y=1}^{N} \exp(Z_x^T \times Z_y)} \tag{10}$$

where, $Z_x^T \in \mathbb{R}^{C\times 1}$ denotes the feature vector of the x^{th} pixel, $SR_{x,y}$ represents the spatial relation among the x^{th} and y^{th} pixels. To preserve the spatial structure, instead of mapping the full row of *SR* into feature space via fully connected (FC) layers (that could distort spatial information), the x^{th} row is reshaped into a two-dimensional spatial map $\mathrm{SR}_{\mathrm{x}}^{\mathrm{row}} \in \mathbb{R}^{\mathrm{H}\times\mathrm{W}}$.

Given that input images typically feature face-centered and structurally consistent content, a lower-resolution representation of pixel-wise spatial relationships is adequate for effective face forgery detection, while also helping to reduce computational overhead.The Spatial Relation Feature $\mathrm{F}_{\mathrm{SRel}}^{\mathrm{x}} \in \mathbb{R}^{\mathrm{C}'\times\mathrm{H}\times\mathrm{W}}$ for the x^{th} pixel is then calculated by applying average pooling and a few convolutional layers to $\mathrm{SR}_{\mathrm{x}}^{\mathrm{row}}$. The final feature map is updated using Equation (11):

$$F_{ud} = \mathcal{I}\left(f_{1\times1}\left(F_{input}\right)F_{SRel}^{x}\right) \tag{11}$$

where, $\mathrm{f}_{1\times1}$ is a 1×1 convolution used to integrate features across channels, and I represents the integration operation of the original and relational features.

This module captures pixel-wise spatial relationships by computing a relation map between all pixel pairs, preserving spatial structure. This helps the model identify subtle inconsistencies in forged images, enhancing forgery detection accuracy.

3.5.2 Regional Deep Forgery Enhancement Module (RDFE): The feature map F_{ud} from the SNRM is given as input to the RDFE along the shallow and deep features to combine the feature from areas with spatial attention strategy to identify the similarity between the features. A fake mask is computed $M_{fake} \in \mathbb{R}^{C\times H\times W}$ by performing pixel-wise classification on F_{ud}. Specifically, F_{ud} which is fed int 1×1 convolutional layer and an adjacent activation layer as in Eq. (12)

$$M_{fake} = Sigmoid\left(\delta_{1\times1}(F_{ud})\right) \tag{12}$$

where, sigmoid $\delta_{1\times1}(F_{ud})$ denotes the output of the convolutional layer, and the sigmoid function squashes each pixel value into the array of [0, 1], representing the likelihood that the equivalent patch is forged.Depending on the M_{fake}, the NRN can highly represent the fake regions effectively. Instead of utilizing M_{fake} as the attention map, the attention map Att_{fake} is generated for the faked region based on the M_{fake} with normalization by Eq. (13)

$$Att_{fake} = \frac{M_{fake}}{\frac{1}{N}\sum_{x=0}^{N} M_{fake}^{x}} \tag{13}$$

In Att_{fake}, the score of every individual pixel correlates positively with the likelihood of being manipulated, thereby directing the model's attention to useful areas. Additionally, there is a significant advantage of Att_{fake} compared to M_{fake}. When the pixel values in M_{fake} are nearly zero, weighting the feature map with M_{fake} results in all position features being suppressed. In contrast, Att_{fake} consistently assigns high weights to regions that exhibit a higher likelihood of influence, regardless of the entire probability values. Finally, the manipulated region feature F_{fake} is computedby,

$$F_{fake} = \frac{1}{N}\sum_{x=0}^{N} Att_{fake}^{x} \times F_{ud}^{x} \tag{14}$$

Also, the attribute of the realarea F_{real} and corresponding attention map Att_{real} are determined in a similiar way where the original mask $M_{real} = 1 - M_{fake}$ is employed instead of M_{fake}.

This module effectively enhances forgery detection by focusing on both manipulated and authentic regions through attention-guided feature aggregation. By generating normalized attention maps from the fake mask, it emphasizes informative regions likely to be forged, enable the approachfor distinguish between actual and duplicate areas with greater precision and robustness.

3.6 Classification layer

To capture multi-scale contextual information, a Spatial Pyramid Pooling (SPP) Layer [32] is employed subsequent to the comprehensive feature extraction and integration procedure within the network. The process aggregates the deep feature maps F_{feat} into containers of varying sizes, concatenating the resulting vectors to produce a fixed-length output appropriate for subsequent fully connected layers and classifiers. This is crucial for maintaining spatial hierarchical structures between features at various scales. SPP layers with L levels are as follows:

$$SPPLayer\ (F_{feat}) = \bigoplus_{l=1}^{L} pool_l\ (F_{feat}) \tag{15}$$

where $\bigoplus$ denotes concatenation procedure, $pool_l\ (F_{feat})$refersto the pooling processinl^{th} level, implementedthrough average pooling in spatial containers of various scales where L is adjusted to be 4 (pools features at four different scales or grid sizes).

The extracted features (F_{fake}, F_{real}) from NRN branch, F_{feat} from SPP layer and LSF derived F_{freq} are fed into the G of SPP which fuses the features as feature vector at different level as by Eq. (16)

$$SPP\ (G) = \text{SPP}(\phi(F_{feat} \oplus F_{freq} \oplus F_{real} \oplus F_{fake})) \tag{16}$$

Subsequently, the classifier performs binary classification (*real or fake*) using dropout layer, FC, and softmax layer, enabling effective forged face detection.

This model accurately detect face forgeries under diverse compression conditions, ranging from highly compressed to low-compression or uncompressed images, demonstrating robustness in real-world deepfake scenarios.

Alogrithm: TNVS-Net for detecting highly compressed, low or uncompressed face forgeries.

Input: Collected facial images that consist of real and fake face images

Output: Detecting face forgeries from all compressed scenerios (Highly compressed, low or uncompressed)

1. Divide the input image into patches of size
2. Compute proximity between image patches using self-attention using Eq. (1)
3. Determine output of query image patch by weighted sum of value embeddings by Eq. (2)
4. Reshape and concatenate outputs from multiple patch scales into a unified representation.
5. Apply 2D-FFT on texture map to obtain frequency domain representation using Eq. (3)
6. Apply learnable frequency filter via element-wise multiplication by Eq. (4)
7. Apply inverse FFT to obtain spatial-domain frequency features using Eq. (5)
8. Compute LFS in defined frequency bands using Eq. (6)
9. Fuse RGB and frequency features using cross-modal multi-head attention using Eq. (7)
10. Concatenate outputs of attention heads as per Eq. (8)
11. Apply 1×1 convolution and residual connection to obtain final fused feature by Eq. (9)
12. Compute spatial relation map using matrix multiplication and softmax as in Eq. (10)
13. Integrate spatial relation feature with input feature map using convolution using Eq. (11)
14. Compute fake mask for manipulated regions using sigmoid activation by Eq. (12)
15. Generate normalized attention map for fake regions byEq. (13)
16. Compute manipulated region features using attention map using Eq. (14)
17. Apply Spatial Pyramid Pooling (SPP) on extracted features by Eq. (15)
18. Concatenate all feature representations and feed into classifier using Eq. (16)
19. Classifier outputs final prediction: real or fake face

4. RESULT AND DISCUSSION

4.1 Dataset Description

Table 1 Dataset Discription

Dataset	Key Characteristics	Manipulation Methods / Formats	Size & Sources
FaceForensics++ [33, 34]	Controlled conditions, multiple compression levels (Raw, HQ, LQ), high-quality manipulations	DeepFakes (DF), Face2Face (F2F), FaceSwap (FS), NeuralTextures (NT)	15,000 original + manipulated videos sourced from 977 YouTube videos (frontal faces)

Celeb-DF (v2) [35, 36]	Realistic DeepFakes, diverse demographics, high visual quality	DeepFake (DF)	590 original + 5,639 DeepFake videos, collected from YouTube
WildDeepfake(DF-W) [37, 38]	Real-world DeepFakes, challenging and unconstrained environments	DeepFake (DF)	7,314 face sequences from 707 real-world DeepFake videos collected online

The collected dataset is originally in video format. For the experimental evaluation of the proposed method, the videos are converted into individual image frames to enable frame-level analysis, facilitate the extraction of spatial features, and support the use of image-based deep learning models for face forgery detection.

4.2 Experimental Settings

To evaluate performance improvements, both existing and proposed models are implemented using Python on three different datasets. Each video is split into image frames, and the training uses 80% of the dataset and testing uses 20%.Table 2 depicts the data split for 80% and 20%

Table 2 Dataset Splitting

Dataset	Train (Real)	Test (Real)	Train (Fake)	Test (Fake)
FaceForensics++	8,000	2,000	16,000	4,000
Celeb-DF	3,556	890	22,556	5,639
DF-W	13,480	3,370	13,592	3,397
Total	**25,036**	**6,260**	**52,148**	**13,036**

Each dataset provides a balanced mix of authentic and manipulated samples to ensure robust model evaluation.

Table 3 List of Hyperparameters for TNVS for Face Forgery Detection

Parameters	Optimal Range
Training rate	0.01
Dropout rate	0.5
No. of epochs	200
Batch size	256
Optimizer	Adam
Loss	Cross-entropy

Table 3 outlines the key hyperparameters selected for training the TNVS (Transformer-based Network for Visual Spoofing) model, which are optimized to enhance the accuracy and stability of face forgery detection during the training process.

4.3 Performance Evaluation Metrics

The model's performance in Automated Spoof Detection (ASD) is evaluated using four key measures:

ACCURACY: It represents the proportion of properly identified samples out of all samples examined.

$$Accuracy = \frac{True\ Positive\ (TP) + True\ Negative\ (TN)}{TP + TN + False\ Positive\ (FP) + False\ Negative\ (FN)}$$

Here, TP denotes to the cases that the model accurately identified that an image is a forgery when it is indeed a forgery. TN denotes to the cases that the model accurately identified that an image is authentic when it is truly authentic. FP denotes the cases where The model mistakenly forecasted that an image is fake while it is real. FN indicates the cases where the model incorrectly predicted that an image is authentic when it is actually a forgery.

PRECISION: It measures the ratio of accuratelyidentified fake (forgery) samples out of all samples the model anticipated as fake.

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

If the model predicts that a sample is fake, precision tells us how many of those predictions were actually correct**.** High precision means few FP (i.e., the model rarely labels a real image as fake).

RECALL: Recall ascertains the percentage of false samples that were accurately detected by the model.

$$Recall = \frac{TP}{TP+FN}$$

It shows how many fake samples the model successfully detected**.** High recall means few FN (i.e., the model rarely misses a fake).

F1-SCORE: It represents the weighted average of Precision and Recall. It effectively combines both metrics, particularly in cases of dissimilar class distribution.

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

It provides a unified metric that accounts for both FP and FN. This is particularly beneficial when precision and recall are both important, such as in deepfake detection, where you want to catch as many fakes as possible without flagging real ones incorrectly**.**

Area Under the Curve (AUC): AUC is a performance metric that measures a model's capacity to differentiate between two categories (e.g., PPD and Non-PPD) at all potential categorization thresholds.

AUC is computed using the Receiver Operating Characteristic (ROC) curve, that is plotted with TP Rate (TPR) against the FP Rate (FPR)

$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{FP + TN}$$

AUC score is obtained by integrating the ROC curve:

$$AUC = \int_0^1 TPR(FPR)\, d(FPR)$$

AUC values range from **0 to 1**

AUC = 1.0 means flawless classification

AUC = 0.5 means random guessing

AUC = nearly 1demonstrates superior model efficiency showing strong discrimination between PPD and Non-PPD.

4.4 Training Vs. Testing Accuracy

An epoch in machine learning represents one full pass through the training dataset. when epochs augments, the model typically improves in learning patterns and generalizing to new data. This chart compares training and testing accuracy at Epochs 20 and 100across three datasets—FaceForensics++**,** Celeb-DF**,** and DF-W

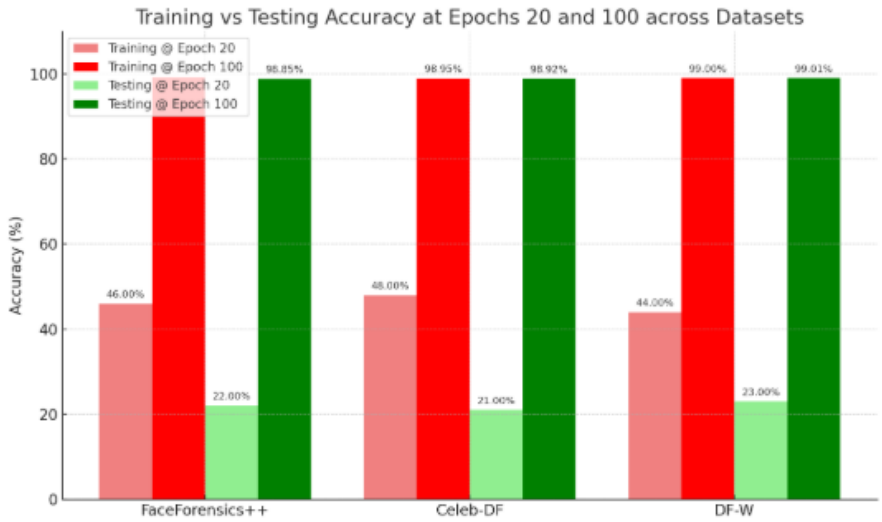


Figure 4 Training vs Testing Accuracy at 20 and 100 across Datasets

Epoch 20: The model has only seen the data 20 times, so it's still in an early learning phase. Accuracy is relatively low (e.g., 46% training and 22% testing on FaceForensics++).

Epoch 100: After 100 passes through the data, the model has learned significantly more. Accuracy is much higher (e.g., ~99% for both training and testing), indicating better performance and generalization.

The training vs. testing accuracy for FaceForensics++, Celeb-DF and DF-W dataset is given in figure 4.For the FaceForensics++ dataset, the training accuracy is 46% at epoch 20 and increases to 99.12% at epoch 100. At epoch 20, the testing accuracy is 22%, but by epoch 100, it rises to 98.85%. For the Celeb-DF dataset, the training accuracy is 48% at epoch 20 and reaches 98.95% at epoch 100. At epoch 20, the testing accuracy is 21%, which improves to 98.92% by epoch 100. For the DF-W dataset, the training accuracy is 44% at epoch 20 and grows to 99% at epoch 100. At epoch 20, the testing accuracy is 23%, and it increases to 99.01% by epoch 100.The model's improved testing accuracy shows its capacity to learn fast and adaptable to novel data.

4.5 Performance Analysis

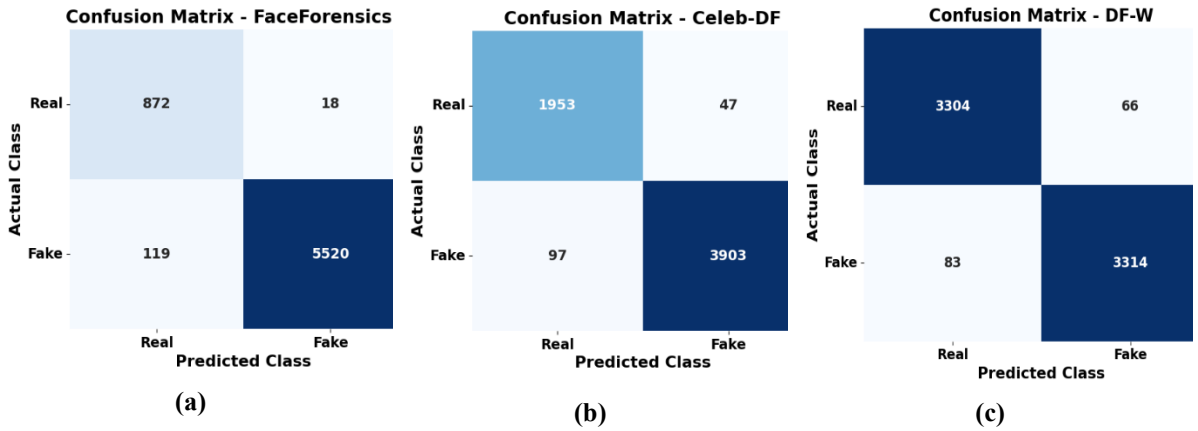


Figure 5 Confusion matrix of testing set for (a) FaceForensics++ (b) Celeb-DF (c) DF-W

In this section, the performance analysis is conducted by comparing the proposed model TNVS-Net with different standard models like VGG19, MesoNet, EfficientNet-B4 and XceptionNet for accuracy, precision, recall and f1-score. The confusion matrix of thesuggested model on testing set is given in figure 5.

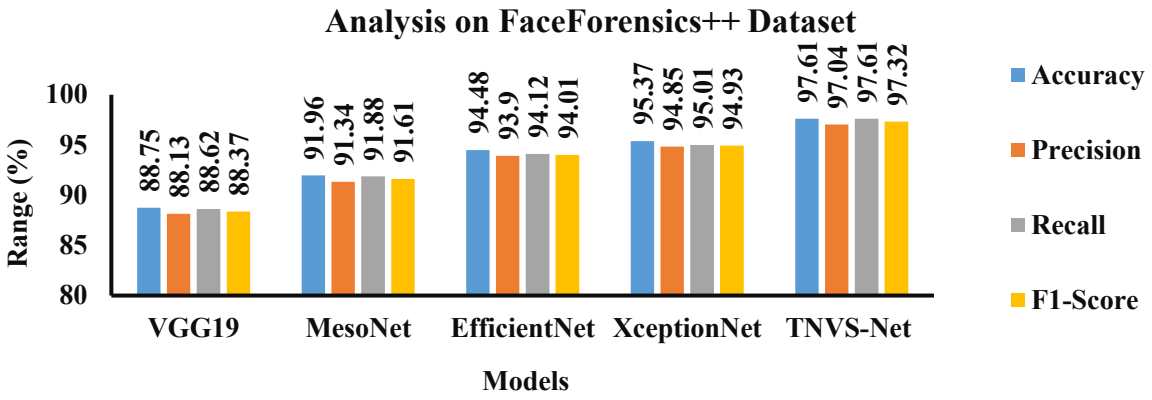


Figure 6 Accuracy Analysis on FaceForensics++ Dataset for proposed and existing models

Figure 6 illustrates the efficiency of several models on the FaceForensics++ dataset for face forgery detection. The proposed TNVS-Net outperforms existing models—VGG19, MesoNet, EfficientNet-B4, and XceptionNet—usingaccuracy, precision, recall, and F1-score**.** Specifically, TNVS-Net achieves up to 9.97% higher accuracy**,** 10.11% higher precision**,** 10.15% higher recall**,** and 10.12% higher F1-score compared to these models.

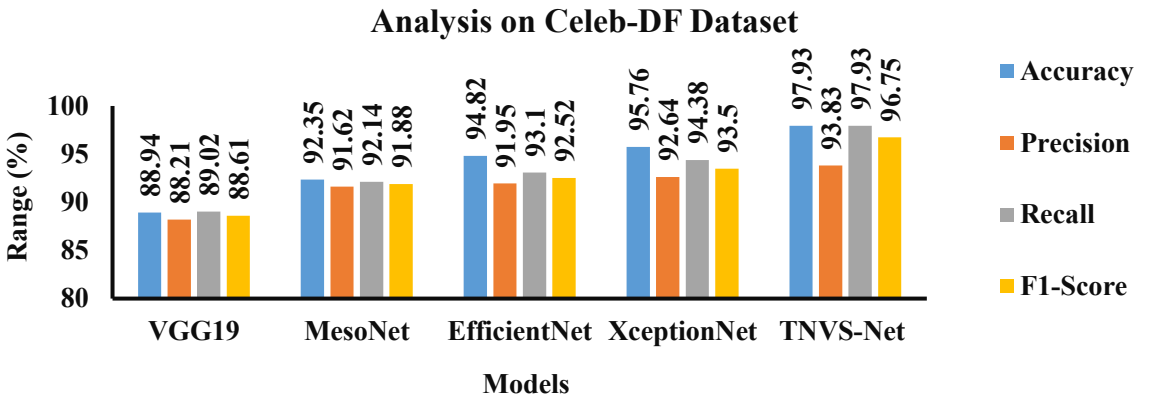


Figure 7 Analysis on Celeb-DF Dataset for proposed and existing models

Figure 7 shows the efficiency of various techniques on the Celeb-DF dataset for face forgery detection. The proposed TNVS-Net outperforms existing models—VGG19, MesoNet, EfficientNet-B4, and XceptionNet—in all key metrics. It achieves up to 10.12% higher accuracy**,** 6.36% higher precision**,** 10.00% higher recall**,** and9.19% higher F1-score compared to these models.

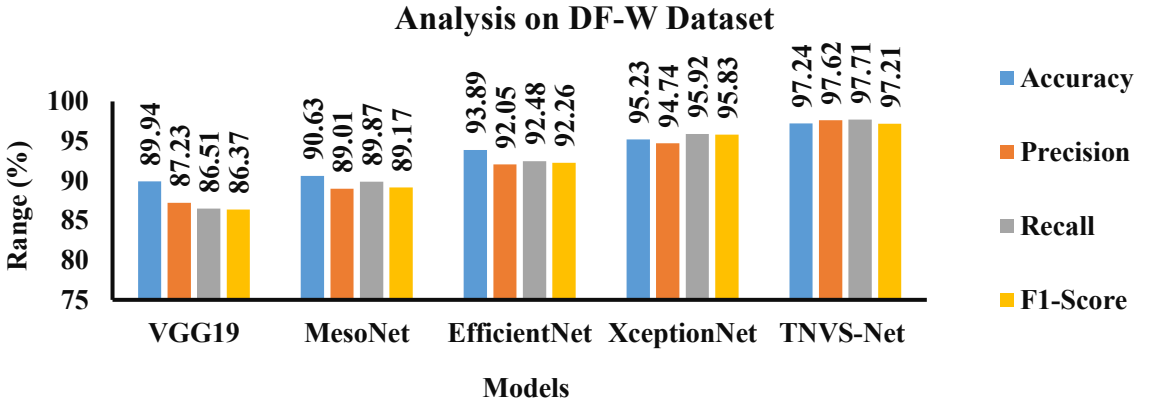
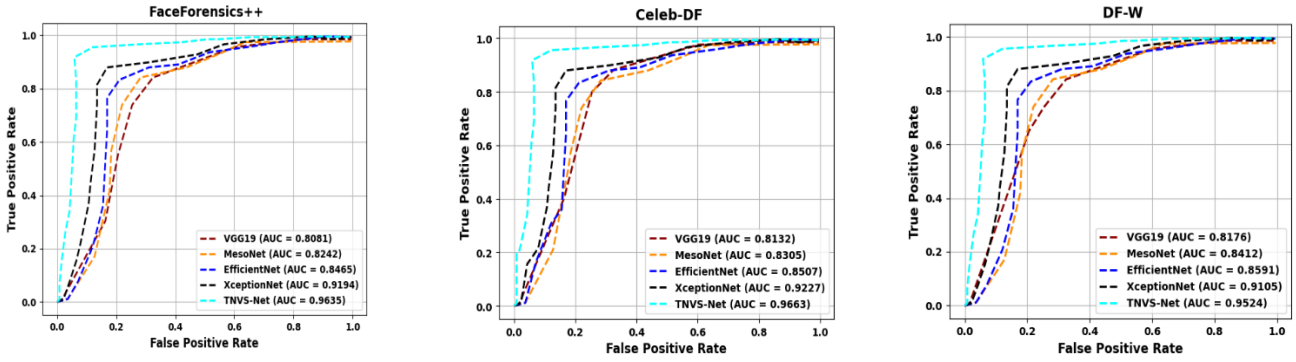


Figure 8 Analysis on DF-W Dataset for proposed and existing models

Figure 8 presents the performance comparison on the DF-W dataset for face forgery detection. The TNVS-Net surpasses all baseline models—VGG19, MesoNet, EfficientNet-B4, and XceptionNet—in accuracy,

precision, recall, and F1-score**.** It shows up to 8.11% higher accuracy**,** 11.92% higher precision**,** 12.96% higher recall**, and** 12.55% higher F1-score over the compared models.



(a) (b) (c)

Figure 9 AUC analysis onFaceForensics++, Celeb-DF and DW-FDataset for proposed and existing models

Figures 9 present the AUC analysis for the proposed TNVS-Net compared against VGG19, MesoNet, EfficientNet-B4, and XceptionNet using the FaceForensics++, Celeb-DF, and DF-W datasets, respectively. Each point on the curve represents the trade-off between correctly identifying forged faces and incorrectly classifying authentic faces as forgeries. The curve's proximity in the top-left corner specifies the superior capability of TNVS-Net to effectively distinguish between forged and authentic face images.

4.6 Computational Metrics

The **TNVS-Net** model demonstrates excellent **computational efficiency** alongside **high detection accuracy** across all datasets

Table 4. Computational efficiency of the proposed model using different datasets

Datasets	Training time (sec)	Testing time (sec)	Detection Accuracy (%)
FaceForensics++	1420	85	98.85
Celeb-DF	1735	92	98.92
DF-W	1260	78	99.01

DF-W achieved the highest accuracy with the lowest computational cost, while Celeb-DF required the most training time, likely due to its complexity. Overall, TNVS-Net effectively balances detection accuracy and computational performance across varying datasets.

4.7 Comparative analysis of feature maps

This analysis compares feature maps generated by different branch strategies of the proposed model across three deep fake datasets. It demonstrates that integrating multiple feature types significantly improves the model's detection accuracy.

Figure 10 Comparison of feature maps between different branch strategies of the proposed model

Figure 10 illustrates a comparative analysis of feature maps derived from different branch strategies in the proposed model, tested on three deepfake datasets: FaceForensics++, Celeb-DF, and DF-W. It highlights the combination of RGB images, frequency domain features, and residual noise patterns through feature fusion improves model detection of modest manipulation cues, improving its overall robustness across varied datasets and forgery techniques.

5. CONCLUSION

In this paper, TNVS-Net is proposed for detecting compressed, low or uncompressed face forgeries. Initially, the pre-trained EfficientNet-B4 backbone is utilized to derive the shallow and deep attributes from the input images. A multi-scale RGB branch utilizes a transformer encoder to capture hierarchical texture representations, while a frequency filter branch applies a 2D-FFT and LFS to expose subtle spectral anomalies indicative of forgery. These spatial and frequency features are fused using a FFM that combines multi-head attention and convolutional refinement to enhance complementarity. To further improve spatial consistency, NRN is introduced, comprising a SNRM to model pixel-wise contextual similarities and a RDFE to localize manipulated regions using an adaptive attention mask. Finally, a SPP layer aggregates multi-scale contextual features into a fixed-length vector, which is classified via fully connected layers and a softmax activation. Experimental results demonstrate the model's superior accuracy compared to classical face forgery detection methods.

References

1. Ballard, L., Lopresti, D., & Monrose, F. (2007). Forgery quality and its implications for behavioral biometric security. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 37(5), 1107-1118.
2. He, Y., Gan, B., Chen, S., Zhou, Y., Yin, G., Song, L., ... & Liu, Z. (2021). Forgerynet: A versatile benchmark for comprehensive forgery analysis. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 4360-4369).
3. Zhuang, W., Chu, Q., Yuan, H., Miao, C., Liu, B., & Yu, N. (2022, July). Towards intrinsic common discriminative features learning for face forgery detection using adversarial learning. In 2022 IEEE International Conference on Multimedia and Expo (ICME) (pp. 1-6). IEEE.
4. Guo, Z., Liu, Y., Zhang, J., Zheng, H., & Shan, S. (2024). Face Forgery Detection with Elaborate Backbone. *arXiv preprint arXiv:2409.16945*.
5. Kwok, A. O., & Koh, S. G. (2021). Deepfake: a social construction of technology perspective. Current Issues in Tourism, 24(13), 1798-1802.
6. Kumar, M., & Sharma, H. K. (2023). A GAN-based model of deepfake detection in social media. Procedia Computer Science, 218, 2153-2162.
7. Sadiq, S., Aljrees, T., & Ullah, S. (2023). Deepfake detection on social media: leveraging deep learning and fasttext embeddings for identifying machine-generated tweets. *IEEE Access*, *11*, 95008-95021.
8. Narayan, K., Agarwal, H., Mittal, S., Thakral, K., Kundu, S., Vatsa, M., & Singh, R. (2022). Desi: Deepfake source identifier for social media. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 2858-2867). Narayan, K., Agarwal, H., Mittal, S., Thakral, K., Kundu, S., Vatsa, M., & Singh, R. (2022). Desi: Deepfake source identifier for social media. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 2858-2867).
9. Negi, S., Jayachandran, M., & Upadhyay, S. (2021). Deep fake: an understanding of fake images and videos. International Journal of Scientific Research in Computer Science Engineering and Information Technology, 7(3), 183-189.

10. Al-Khazraji, S. H., Saleh, H. H., Khalid, A. I., & Mishkhal, I. A. (2023). Impact of deepfake technology on social media: Detection, misinformation and societal implications. The Eurasia Proceedings of Science Technology Engineering and Mathematics, 23, 429-441.
11. Li, J., Xie, H., Li, J., Wang, Z., & Zhang, Y. (2021). Frequency-aware discriminative feature learning supervised by single-center loss for face forgery detection. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 6458-6467).
12. Afchar, D., Nozick, V., Yamagishi, J., & Echizen, I. (2018, December). Mesonet: a compact facial video forgery detection network. In 2018 IEEE international workshop on information forensics and security (WIFS) (pp. 1-7). IEEE.
13. Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., & Nießner, M. (2019). Faceforensics++: Learning to detect manipulated facial images. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 1-11).
14. Cozzolino, D., Thies, J., Rössler, A., Riess, C., Nießner, M., & Verdoliva, L. (2018). Forensictransfer: Weakly-supervised domain adaptation for forgery detection. arXiv preprint arXiv:1812.02510.
15. Du, M., Pentyala, S., Li, Y., & Hu, X. (2020, October). Towards generalizable deepfake detection with locality-aware autoencoder. In Proceedings of the 29th ACM international conference on information & knowledge management (pp. 325-334).
16. Nguyen, H. H., Yamagishi, J., & Echizen, I. (2024, September). Exploring self-supervised vision transformers for deepfake detection: A comparative analysis. In 2024 IEEE International Joint Conference on Biometrics (IJCB) (pp. 1-10). IEEE.
17. Chen, C. F. R., Fan, Q., & Panda, R. (2021). Crossvit: Cross-attention multi-scale vision transformer for image classification. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 357-366).
18. J. Wang, Z. Wu, W. Ouyang, X. Han, J. Chen, Y.-G. Jiang, and S.-N. Li, ''M2TR: Multi-modal multi-scale transformers for Deepfake detection,'' in Proc. Int. Conf. Multimedia Retr., Jun. 2022, pp. 615–623
19. Y. Qian, G. Yin, L. Sheng, Z. Chen, and J. Shao, ''Thinking in frequency: Face forgery detection by mining frequency-aware clues,'' in Proc. 16th Eur. Conf. Comput. Vis. Cham, Switzerland: Springer, 2020, pp. 86–103.
20. Zhou, J., Zhao, X., Xu, Q., Zhang, P., & Zhou, Z. (2024). MDCF-Net: Multi-Scale Dual-Branch Network for Compressed Face Forgery Detection. IEEE Access, 12, pp. 58740 – 58749.
21. Tran, V. N., Kwon, S. G., Lee, S. H., Le, H. S., & Kwon, K. R. (2022). Generalization of forgery detection with meta deepfake detection model. IEEE Access, 11, 535-546.
22. Sun, Y., Zhang, Z., Echizen, I., Nguyen, H. H., Qiu, C., & Sun, L. (2023). Face forgery detection based on facial region displacement trajectory series. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 633-642).
23. Alnaim, N. M., Almutairi, Z. M., Alsuwat, M. S., Alalawi, H. H., Alshobaili, A., & Alenezi, F. S. (2023). DFFMD: a deepfake face mask dataset for infectious disease era with deepfake detection algorithms. *IEEE Access*, *11*, 16711-16722.
24. Ramadhani, K. N., Munir, R., & Utama, N. P. (2024). Improving Video Vision Transformer for Deepfake Video Detection using Facial Landmark, Depthwise Separable Convolution and Self Attention. *IEEE Access*.
25. Liu, Q., Xue, Z., Liu, H., & Liu, J. (2024). Enhancing Deepfake Detection With Diversified Self-Blending Images and Residuals. *IEEE Access*.
26. Zhou, Q., Zhou, Z., Bao, Z., Niu, W., & Liu, Y. (2024). IIN-FFD: Intra-Inter Network for Face Forgery Detection. *Tsinghua Science and Technology*, *29*(6), 1839-1850.
27. Wang, Y., Fu, H., & Wu, T. (2024). Research on the Face Forgery Detection Model Based on Adversarial Training and Disentanglement. *Applied Sciences*, *14*(11), 4702.
28. Khormali, A., & Yuan, J. S. (2024). Self-supervised graph Transformer for deepfake detection. *IEEE Access*.
29. Soudy, A. H., Sayed, O., Tag-Elser, H., Ragab, R., Mohsen, S., Mostafa, T., ... & Slim, S. O. (2024). Deepfake detection using convolutional vision transformers and convolutional neural networks. Neural Computing and Applications, 36(31), 19759-19775.
30. Siddiqui, F., Yang, J., Xiao, S., & Fahad, M. (2025). Enhanced deepfake detection with DenseNet and Cross-ViT. Expert Systems with Applications, 267, 126150.
31. Rao, Y., Zhao, W., Zhu, Z., Lu, J., & Zhou, J. (2021). Global filter networks for image classification. Advances in neural information processing systems, 34, 980-993.
32. He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE transactions on pattern analysis and machine intelligence, 37(9), 1904-1916.
33. https://paperswithcode.com/dataset/faceforensics-1
34. https://www.kaggle.com/datasets/hungle3401/faceforensics
35. https://paperswithcode.com/dataset/celeb-df
36. https://www.kaggle.com/datasets/reubensuju/celeb-df-v2
37. Dolhansky, B., Bitton, J., Pflaum, B., Lu, J., Howes, R., Wang, M., & Ferrer, C. C. (2020). The deepfake detection challenge (dfdc) dataset. arXiv preprint arXiv:2006.07397.
38. https://ai.meta.com/datasets/dfdc/
39. Zi, B., Chang, M., Chen, J., Ma, X., & Jiang, Y. G. (2020, October). Wilddeepfake: A challenging real-world dataset for deepfake detection. In *Proceedings of the 28th ACM international conference on multimedia* (pp. 2382-2390).
40. https://paperswithcode.com/dataset/wilddeepfake