Improved Image Forgery Detection Based on VGG16, Cosine Similarity, and Support Vector Machines

Issam SHALLAL
University of Sousse,
ISITCom, LATIS- Laboratory
of Advanced Technology and
Intelligent Systems, 4002, Sousse, Tunisia
University of Anbar
esam.khamis@uoanbar.edu.iq

Lamia RZOUGA HADDADA
University of Sousse,
Institut Supérieur des Sciences
Appliquées et de Technologie de Sousse
LATIS- Laboratory of Advanced
Technology and Intelligent Systems, 4002,
Sousse, Tunisia
lamia.rzouga@issatso.u-sousse.tn

Najoua ESSOUKRI BEN AMARA
University of Sousse,
Ecole Nationale d'Ingénieurs de Sousse,
LATIS- Laboratory of
Advanced Technology and
Intelligent Systems, 4002,
Sousse, Tunisia
najoua.benamara@eniso.rnu.tn

***Abstract*—Image forgery detection is crucial in digital forensics, cybersecurity, and legal investigations. Despite advancement, detecting subtle manipulations like copy-move forgeries remains challenging due to increasingly realistic images. This paper proposes a hybrid approach that combines a pretrained VGG16 model for feature extraction, cosine similarity for block-level comparison, and support vector machines for classification, addressing key limitations of existing methods. Through a comparative evaluation of CNN architectures, VGG16 is identified as the most effective for extracting discriminative features in this context. Cosine similarity quantifies the similarity between image block features to enable the model to focus more effectively on the tampered regions that closely resemble the original ones, and SVMs are leveraged to classify authentic versus forged regions. This novel integration of deep learning for feature extraction and classical classification techniques is highly accurate with minimal false positives, without relying on handcrafted features. Experiments on the MICC-F2000 dataset demonstrate the method's strong performance, achieving 99.59% precision, 98.00% recall, and a 0.99 F1-score.**

***Index Terms*—Copy-move forgery detection, Hybrid-based techniques, VGG16, Cosine similarity, Support Vector Machines.**

I. Introduction

Copy-move forgery (CMF) is among the most prevalent techniques used to manipulate images. It is often employed to alter visual evidence or mislead viewers by concealing certain elements or duplicating key objects within an image In an era where digital image manipulation has become increasingly accessible and sophisticated [1], [2], Copy-Move Forgery Detection (CMFD) has emerged as a vital area of research with diverse applications in digital forensics, content verification, and authenticity preservation [3], [4]. CMFD is particularly crucial in forensic investigations, where altered images may be used as evidence in criminal proceedings. In journalism, it ensures the integrity of visual content, safeguarding readers from manipulated and deceptive imagery.

Social media platforms use CMFD to detect and remove forged images, helping combat misinformation and maintain trust. In e-commerce, CMFD authenticates product images, ensuring transparency. In digital art, it verifies artwork authenticity, reducing counterfeit proliferation. Across law enforcement, journalism, social media, commerce. CMFD is vital for preserving the credibility and reliability of digital images in an age of widespread manipulation [5] .

Despite advancements, CMFD remains a challenging task, particularly when confronted with subtle and expertly executed forgeries. Traditional handcrafted algorithms, which rely on manually defined features, often fall short due to their rigid assumptions and vulnerability to transformations such as rotation, scaling, and blending [6], [7]. These manipulations make forged regions nearly indistinguishable from authentic ones, posing significant hurdles for detection methods [5]. On the other hand, fully end-to-end CNN models may lack the explicit discriminative mechanisms needed to distinguish between highly similar tampered and original regions. To address this, we incorporate cosine similarity to effectively measure feature-level resemblance, and leverage Support Vector Machines (SVMs) for its robust decision boundaries, offering better generalization in challenging forgery detection scenarios.

The main contributions of this study are as follows:

- A comprehensive evaluation and comparative analysis of multiple pretrained CNN models, including VGG16, ResNet18, DenseNet121, and AlexNet to determine the most effective feature extractor for sophisticated CMFD.
- The design and integration of a novel hybrid detection framework that combines VGG16 for feature extraction, cosine similarity for measuring inter-block similarity while concentrating on tampered regions resembling the original ones, and SVMs for classification.

The rest of the paper is structured as follows: Section II reviews relevant literature. Section III outlines the proposed methodology. Section IV presents the experimental results and analysis. Section V concludes the paper with insights and future research directions.

II. Related work

CMF in digital images is a complex and demanding task that requires advanced tools and techniques. Over the years, researchers have developed a wide range of methods to tackle this challenge [5]. These approaches can be broadly categorized into traditional CMFD methods (such as block-based and keypoint-based techniques), hybrid strategies, and those leveraging deep learning frameworks.

A. Conventional CMFD techniques

Methods for detecting forgery are generally classified into block-based and keypoint-based types, typically involving feature extraction, matching, and forgery localization.

- Block-based methods: These divide the image into overlapping or non-overlapping blocks. Features are extracted using techniques like discrete cosine transform, principal component analysis, or local binary pattern and matched using correlation or Euclidean distance. Forgery localization often uses geometric transformations with RANSAC to filter mismatches [8]. However, they are computationally intensive and struggle with geometric transformations.
- Keypoint-based methods: They detect keypoints (e.g.corners and edges) using algorithms like Scale-Invariant Feature Transform (SIFT) or Speeded Up Robust Features (SURF) [9], [10], and match them using clustering or nearest-neighbor techniques. Though efficient and robust to transformations, they face challenges in uniform regions or misclassifying similar images as forgery.

Despite their effectiveness, conventional methods have limitations, including manual parameter adjustment, dataset dependence, and reduced generalizability.

B. Hybrid CMFD techniques

Hybrid methods [12], [13] are particularly effective for identifying diverse CMF, handling various transformations, and managing complex backgrounds. For example, in [12], the authors proposed a forensic framework that combined adaptive and hybrid strategies, incorporating Haar discrete wavelet transforms, SURFs, histogram of oriented gradients, and a probabilistic filter for classification.

C. Deep-learning based CMFD methods

Recent deep learning methods have significantly advanced forgery detection in computer vision by automatically extracting hierarchical features, eliminating the need for manual feature engineering. However, their reliance on large datasets remains a challenge. Several solutions have addressed this, as outlined below:

- **CMFD using CNNs**: Researchers have tailored deep learning models [14] like AlexNet,VGG16, VGG19, ResNet, GoogleNet, and DenseNet for forgery detection by fine-tuning their layers and training them on domain-specific datasets [14], [15]. While these architectures have been adapted to effectively detect copy-move, splicing, and inpainting forgery, their primary strength lies in feature extraction and classification rather than precise pixel-level localization.
- CMFD using object detection networks: Frameworks like R-CNN, Faster R-CNN and Mask R-CNN have been modified to identify forgery regions by adjusting their layers and training them with specialized datasets [16], [17].
- CMFD using autoencoders: An autoencoder compresses and reconstructs an image to detect inconsistencies, using symmetric or asymmetric layers to enhance performance. Rather than reconstructing the full image, it can output a binary mask to localize forged and authentic pixels, enabling effective detection of various forgery types [18], [19].
- CMFD using Generative Adversarial Networks (GANs): GANs use a generator to create samples resembling training data and a discriminator to assess authenticity. Some GAN-based methods employ one-class classification or combine sparse autoencoders with SVMs for forgery detection [7], [20].
- CMFD using Recurrent Neural Networks (RNNs): RNNs, particularly LSTMs, capture spatial dependencies and are used in combination with CNNs and autoencoders for pixel-level forgery detection [4], [21].

A summary of these models and their performance is provided in Table I. Despite advancements, challenges in CMF detection persist [3]. One major issue is generalizability, as models often struggle with unfamiliar data or unclear copied regions. Another challenge is background blending, where forgeries seamlessly integrate into complex backgrounds, complicating detection. These challenges underscore the need for improved methods to handle such complexities in real-world data.

III. Methodology

This methodology targets CMFD by classifying image blocks as authentic or forged through a three-stage process: (1) deep feature extraction using a pre-trained CNN, (2) block similarity assessment via cosine similarity, and (3) final classification using an SVM. To identify the most effective feature extractor, we conduct a comparative evaluation of four widely used pre-trained CNN architectures (VGG16, ResNet18, DenseNet121, and AlexNet) on a benchmark dataset. VGG16 is selected based on superior performance across precision, recall, and F1-score metrics. While the individual components are standard, their integration in a block-wise forgery detection pipeline and extensive comparative validation contribute to the robustness and novelty of this approach.

VGG16 is known for its simple, uniform architecture with 16 layers, utilizing 3x3 convolutional filters and max-pooling, although it is computationally heavy due to its large number of parameters. ResNet18, part of the ResNet family, introduces residual connections, allowing the model to train deeper networks by addressing the vanishing gradient problem, making it

TABLE I
SUMMARY OF STUDIES ON CMFD EMPLOYING DEEP LEARNING-BASED METHODS

DL architectures	Ref	Techniques	Datasets	Performances
CNN	[15]	Dual-branch CNN	MICC-F2000	Robust to scaling
	[14]	AlexNet and logistic regression	MICC-F600, MICC-F2000	Improving accuracy on smaller datasets
RNN	[21]	CNN with LSTM	NIST16, COVERAGE	Able to detect all types of forgery
	[4]	CNN with ConvLSTM	MICC-F220, MICC-F600, MICCF2000	Hybrid layers enhancing performance, dataset combination improves generality
GAN	[20]	GAN with SVM	MICC-F600, CoMoFod	Necessitating a large amount of data
	[7]	Dual-order attentive GAN	CASIA, CoMoFod	Robust against geometric transformations
Autoencoders	[18]	LSTM and rotating residual units	COVERAGE, NIST16, CASIA	Detection of all types of forgery
	[19]	U-net (encoder(ResNet)-decoder)	COVERAGE, NIST16, CASIA, CoMoFod	Detection of all types of forgery images with high computational complexity
Object detection	[17]	Mobilenet with mask R-CNN	MICC-F220, MICC-F600, and MICCF2000	Robust to detect forgery
	[22]	DenseNet-41 with mask R-CNN	CoMoFod, MICC-F2000, CASIA V2	Robust against geometric transformations

faster and more efficient than other models. DenseNet121, with 121 layers, uses densely connected blocks where each layer receives input from all previous layers, improving feature reuse and reducing the parameter count. AlexNet, one of the first deep learning models to perform well on ImageNet, consists of 8 layers and introduces key innovations like ReLU activations and dropout, although its architecture is relatively shallow by modern standards. In this approach, cosine similarity is used to measure the similarity between image blocks, which is a key step in detecting image forgery. However, we incorporate an SVM to transform this similarity score into a robust binary classification, distinguishing between "authentic" and "forged" blocks, to handle complexities and variations in the data. In practice, we treat cosine similarity as a feature for the SVM, rather than a final decision-making criterion. This allows the SVM to learn from the similarity values and make more accurate classifications. This approach not only improved the flexibility of the system but also its performance, especially when dealing with diverse and complex datasets.

The methodology of this approach is structured as follows:

A. Image preprocessing

- The image is resized to 224×224 pixels, which is the standard input size for VGG16.
- The pixel values are normelized, typically by subtracting the mean RGB values used during the pre-training of VGG16, or by dividing by 255 to scale them to the range $[0, 1]$.
- Each image is divided into overlapping/non-overlapping blocks of size 32×32. With a stride of x pixels (e.g. 32 for non-overlapping or 16 for 50% overlap), this results in approximately 49 blocks per image. Each block is then passed through the CNN for feature extraction.

B. Feature extraction using VGG16 (FC7 Layer)

- The image is passed through the VGG16 network, which is pre-trained on ImageNet. The network consists of several convolutional layers followed by fully connected layers.
- For feature extraction, the output from the FC7 layer (second-to-last fully connected layer before softmax output) is used. This layer produces a 4096-dimensional feature vector for each image or block (equation 1).

The extracted feature vector for an image or block is:

$$F_{\text{image}} = [f_1, f_2, \ldots, f_{4096}]^T \tag{1}$$

where $F_{\text{image}} \in \mathbb{R}^{4096}$ is the feature vector representing the visual information of the image or the extracted block. To capture local patterns, each image is partitioned into fixed-size blocks (32×32). A total of 49 blocks are extracted per image, and each is processed through the CNN [23] to obtain its feature representation.

C. Using cosine similarity to compare blocks

When working with image blocks (e.g. using a sliding window), each block extracted from the image is converted into a feature vector from the FC7 layer of VGG16.

1) Cosine similarity calculation: Once you have the feature vectors of blocks B_i and B_j, the cosine similarity [24] is computed between the feature vectors F_{B_i} and F_{B_j}(equation 2):

$$\text{Sim}_{\cos}(B_i, B_j) = \frac{F_{B_i} \cdot F_{B_j}}{\|F_{B_i}\| \|F_{B_j}\|} \tag{2}$$

where: F_{B_i} and F_{B_j} are the feature vectors of blocks B_i and B_j, $\cdot$ denotes the dot product and $\| \cdot \|$ denotes the L_2-norm (Euclidean norm) of the vector.

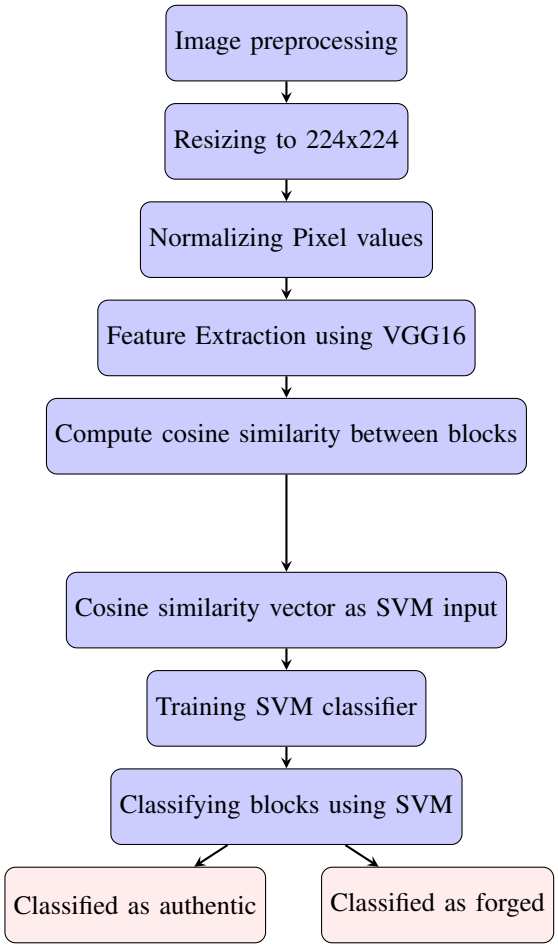

Fig. 1. Block diagram of image forgery detection process using VGG16, cosine similarity, and SVM

2) Similarity thresholding: If the cosine similarity [24] between two blocks exceeds a predefined threshold θ (e.g., $\theta = 0.9$), these blocks are considered *forged* (copied and moved from another region of the image). Otherwise, they are considered *authentic* (equation 3).

$$\text{If } \text{Sim}_{\cos}(B_i, B_j) \geq \theta \quad \text{then} \quad B_i, B_j \text{ are forged.} \tag{3}$$

D. SVM classifier training

After calculating the cosine similarity between blocks, these similarity values are used as input to train the SVM [25]. Here is a detailed breakdown of the process:

1) Labeling Data: For each pair of blocks B_i and B_j, a label is assigned:

- 1 (forged) if $\text{Sim}_{\cos}(B_i, B_j) \geq \theta$,
- 0 (authentic) if $\text{Sim}_{\cos}(B_i, B_j) < \theta$.

2) SVM training: The labeled dataset (cosine similarities and corresponding labels) to train an SVM classifier. The input to the SVM will be the cosine similarity between feature vectors of blocks, and the output will be a binary classification: authentic (0) or forged (1).

The SVM learning algorithm seeks to maximize the margin γ between the two classes by solving the following optimization problem (equation 4):

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 \quad \text{subject to} \quad y_i\left(\mathbf{w} \cdot x_i + b\right) \geq 1, \quad \forall i \tag{4}$$

where: $\mathbf{w}$ is the weight vector of the hyperplane, b is the bias term, x_i is the cosine similarity value for block B_i and y_i is the label associated with B_i (0 for authentic and 1 for forged).

A Radial basis function kernel is utilized due to its ability to handle non-linear relationships in the feature space. The SVM is trained with the regularization parameter $C = 1.0$ and the kernel coefficient $\gamma = 0.01$, selected via cross-validation.

E. Classifying blocks using the SVM

Once the SVM is trained, it can be used to classify new blocks extracted from an image. Each block is compared to other blocks to calculate its cosine similarity, and these similarity values are fed into the SVM classifier.

- If the SVM predicts 1 for a block pair, then those blocks are classified as *forged*.
- If the SVM predicts 0, the blocks are classified as *authentic*.

IV. Expreiments

This section outlines the environmental setup for training and testing employed in the proposed method and evaluates its performance compared to current state-of-the-art techniques.

A. Dataset

To train and test models, deep learning frameworks require large datasets. This study uses the MICC-F220, the MICC-F2000 [26], CASIA 1.0 and the CASIA 2.0 [27] datasets. Table II provides all the information about these databases.

TABLE II
Image forgery database specifications

	Authentic images	Forged image	Total number of images
MICCC-F220	110	110	220
MICCC-F2000	1,300	700	2,000
CASIA 1.0	800	925	1,721
CASIA 2.0	7,491	5,123	12,614

Each dataset is divided into 80% for training and 20% for testing.

B. Evaluation metrics

To assess the performance of the model, standard evaluation metrics can be employed, including [5]:
Precision: The ratio of correctly predicted positive instances to the total predicted positives:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5}$$

Recall: The ratio of correctly identified positive instances to the total actual positives:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6}$$

F1-score: The harmonic mean of precision and recall, providing a balanced measure of both:

$$\text{F1-score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{7}$$

Accuracy: The ratio of correct predictions to the total number of predictions made:

$$\text{Acc} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{8}$$

C. Results

The performance evaluation, presented in Table III, reveals that VGG16 outperforms the other models, achieving the highest accuracy. This superior performance suggests that VGG16's deeper architecture allows it to capture and process more complex features, making it particularly effective for the given task. Additionally, when compared to current state-of-the-art approaches, VGG16 demonstrates competitive or even better results, reinforcing its suitability for similar applications. After selecting VGG16, we evaluate the proposed approach

TABLE III
PERFORMANCE METRICS OF DIFFERENT DEEP LEARNING MODELS ON THE MICC-F220, MICC-F2000, CASIA-1, AND CASIA-2 DATASETS

Dataset	Deep learning model	Acc	Precision		Recall		F1-score	
			A	F	A	F	A	F
MICC-F220	VGG16	**0.93**	**1.0**	**0.88**	**0.86**	**1.0**	**0.93**	**0.94**
	Resnet18	0.91	1.0	0.85	0.82	1.0	0.90	0.92
	Alexnet	0.89	1.0	0.81	0.77	1.0	0.87	0.90
	Densenet121	0.91	1.0	0.85	0.82	1.0	0.90	0.92
MICC-F2000	VGG16	**0.90**	**0.88**	**0.94**	**0.97**	**0.76**	**0.93**	**0.84**
	Resnet18	0.89	0.88	0.91	0.96	0.76	0.92	0.83
	Alexnet	0.88	0.88	0.88	0.95	0.76	0.91	0.82
	Densenet121	0.90	0.91	0.90	0.97	0.76	0.92	0.84
CASIA-1	VGG16	**0.91**	**0.87**	**0.92**	**0.96**	**0.82**	**0.92**	**0.88**
	Resnet18	0.88	0.89	0.89	0.94	0.81	0.91	0.87
	Alexnet	0.85	0.84	0.85	0.91	0.79	0.88	0.83
	Densenet121	0.89	0.90	0.88	0.95	0.78	0.91	0.84
CASIA-2	VGG16	**0.90**	**0.90**	**0.92**	**0.97**	**0.80**	**0.93**	**0.87**
	Resnet18	0.87	0.88	0.89	0.95	0.75	0.89	0.82
	Alexnet	0.85	0.82	0.84	0.90	0.79	0.86	0.81
	Densenet121	0.88	0.88	0.87	0.94	0.76	0.89	0.83

on the MICC-F2000 dataset. The results in Table IV and the confusion matrix in Figure 2 demonstrate its effectiveness in detecting digital forgery. Key findings show that the method provides:

- Balanced precision and recall: Effective detection with minimal false positives.
- Robustness: Handling a range of forgery types, from subtle to overt.
- Low false positives: Ensures authentic images are not wrongly flagged.

The experimental results presented in Table IV, along with the confusion matrix in Figure 2, on the MICC-F2000 dataset, clearly demonstrate the effectiveness of the suggested method in detecting digital forgery. These findings highlight

TABLE IV
PERFORMANCE OF PROPOSED APPROACH

Metric	Value
Number of forged images	492
Number of authentic images	508
Accuracy	98.80%
Precision	99.59%
Recall	98.00%
F1-Score	0.99
Total predictions	1000
Correct predictions	988
Incorrect predictions	12

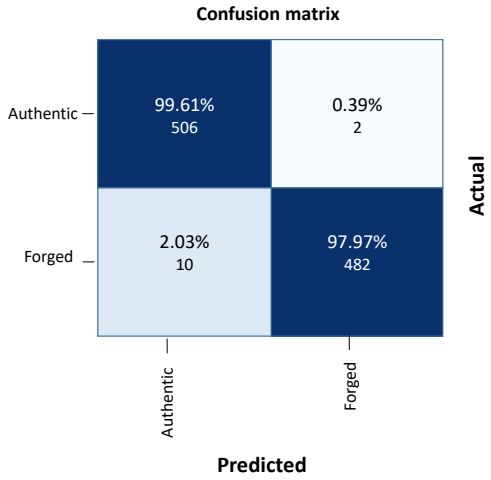

Fig. 2. Confusion matrix of proposed approach

the robustness and accuracy of the approach in distinguishing between authentic and manipulated images, underscoring its potential for practical applications in digital forensics.

TABLE V
PERFORMANCE ASSESSMENT OF PROPOSED APPROACH IN COMPARISON TO RELATED WORK ON MICCF2000

Ref	Year	Methods	Performances
[28]	2021	Smaller VGGNet and MobileNet	Acc=85%
[14]	2022	Pre-trained AlexNet model	Acc=94%
[29]	2022	Hybrid deep learning model	Acc=95%
[30]	2023	CNN	Acc=89%
[6]	2023	PointRend-RegNetX	Acc=86.4%
[31]	2024	Combined SIFT and RVM model	Acc=94%
[9]	2024	Combined MobileNetV2, PCA and random forest	Acc=96.75%
[8]	2024	Combined MobileNetV2, PCA and random forest	Acc=96.37%
Proposed approach	**2025**	**Hybrid approach based on Vgg16, cosine similarity and SVM**	**Acc=98.80%**

Furthermore, Table V presents a comparison of the accuracy achieved by the suggested approach with various state-of-the-art methods [6], [14], [28]–[31] for the detection of digital

image forgery. By leveraging complementary strengths, the proposed method significantly improves detection accuracy and demonstrates strong adaptability to evolving forgery techniques.

V. Conclusion

This study has put forward a hybrid image forgery detection method that combines VGG16-based deep feature extraction, cosine similarity for block comparison and SVM for classification. Among four evaluated CNN models (VGG16, ResNet18, DenseNet121 and AlexNet) VGG16 has achieved the highest accuracy. Cosine similarity has improved the model sensitivity to subtle forgeries, resulting in a precision of 99.59%, a recall of 98.00%, an F1-score of 0.99 and accuracy of 98.80% on the MICC-F2000 dataset. While the method integrates well-known components, its contribution lies in their effective combination and empirical validation, offering a competitive and robust solution to complex forgery detection. Future work will explore forgery type identification and lightweight, interpretable models to further enhance practical applicability.

References

[1] E. Liang, K. Zhang, Z. Hua, Y. Li, and X. Jia, "TransCMFD: An adaptive transformer for copy-move forgery detection," *Neurocomputing*, vol. 130110, 2025, Elsevier.

[2] L. Rzouga Haddada, B. Dorizzi, and N. Essoukri Ben Amara, "Watermarking signal fusion in multimodal biometrics," in *Proc. International Image Processing, Applications and Systems Conference*, pp. 1–6, IEEE, 2014.

[3] N. A. M. Abir, N. B. Abd Warif, and N. Zainal, "An automatic enhanced filters with frequency-based copy-move forgery detection for social media images," *Multimedia Tools and Applications*, vol. 83, no. 1, pp. 1513-1538, 2024.

[4] M. A. Elaskily, M. H. Alkinani, A. Sedik, and M. M. Dessouky, "Deep learning based algorithm (ConvLSTM) for copy move forgery detection," *Journal of Intelligent Fuzzy Systems*, vol. 40, no. 3, pp. 4385–4405, 2021.

[5] S. Gazzah, L. Rzouga Haddada, I. Shallal, and N. Essoukri Ben Amara, "Digital Image Forgery Detection with Focus on a Copy-Move Forgery Detection: A Survey," in *2023 International Conference on Cyberworlds (CW)*, 2023, pp. 240–247. IEEE.

[6] M. H. Farhan, K. Shaker, and S. Al-Janabi, "Efficient approach for the localization of copy-move forgeries using PointRend with RegNetX," *Baghdad Sci. J.*, vol. 21, no. 4, pp. 1416, 2024.

[7] A. Islam, C. Long, A. Basharat, and A. Hoogs, "DOA-GAN: Dual-order attentive generative adversarial network for image copy-move forgery detection and localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 4676–4685.

[8] I. Shallal, L. Rzouga Haddada, and N. Essoukri Ben Amara, "Lightweight Hybrid Model Combining MobileNetV2 and PCA for Copy-Move Forgery Detection," in *Proc. 2024 17th Int. Conf. on Development in eSystem Engineering (DeSE)*, pp. 107–112, IEEE, 2024

[9] I. Shallal, L. Rzouga Haddada, and N. Essoukri Ben Amara, "Enhanced Detection of Copy-Move Forgery by Fusing Scores From Handcrafted and Deep Learning-Based Detection Systems," in *Proc. 2024 17th Int. Conf. on Development in eSystem Engineering (DeSE)*, pp. 113–118, IEEE, 2024.

[10] I. Hamrouni Trimech, N. Messaoudi, and N.Essoukri Ben Amara, "A novel feature combination approach for driver fatigue detection," in *Proceedings of the 2022 19th International Multi-Conference on Systems, Signals & Devices (SSD)*, 2022, pp. 475–479.

[11] C.-C. Chen, W.-Y. Lu, and C.-H. Chou, "Rotational copy-move forgery detection using SIFT and region growing strategies," *Multimedia Tools and Applications*, vol. 78, pp. 18293–18308, 2019.

[12] M. M. A. Alhaidery and A. H. Taherinia, "A passive image forensic scheme based on adaptive and hybrid techniques," *Multimedia Tools and Applications*, vol. 81, pp. 12681–12699, 2022.

[13] M. M. A. Alhaidery, A. H. Taherinia, and H. S. Yazdi, "Cloning detection scheme based on linear and curvature scale space with new false positive removal filters," *Multimedia Tools and Applications*, vol. 81, pp. 8745–8766, 2022.

[14] B. T. Hammad, I. T. Ahmed, and N. Jamil, "A secure and effective copy-move detection based on pretrained model," in *2022 IEEE 13th Control and System Graduate Research Colloquium (ICSGRC)*, 2022, pp. 66–70. IEEE.

[15] A. Goel, M. Rathi, P. Sharma, and L. Singh, "Dual-Channel Convolutional Networks for Image Classification," *International Journal of Computer Vision*, vol. 129, no. 6, pp. 983-997, 2021.

[16] X. Wang, H. Wang, S. Niu, J. Zhang, *et al.*, "Detection and localization of image forgeries using improved mask regional convolutional neural network," *Mathematical Biosciences and Engineering*, vol. 16, no. 5, pp. 4581–4593, 2019, American Institute of Mathematical Sciences (AIMS).

[17] K. D. Kadam, S. Ahirrao, and K. Kotecha, "Efficient Approach towards Detection and Identification of Copy Move and Image Splicing Forgeries Using Mask R-CNN with MobileNet V1," *Computational Intelligence and Neuroscience*, vol. 2022, no. 1, p. 6845326, 2022, Wiley Online Library.

[18] H. Chen, C. Chang, Z. Shi, and Y. Lyu, "Hybrid features and semantic reinforcement network for image forgery detection," *Multimedia Systems*, vol. 28, no. 2, pp. 363–374, 2022.

[19] F. Z. El Biach, I. Iala, H. Laanaya, and K. Minaoui, "Encoder-decoder based convolutional neural networks for image forgery detection," *Multimedia Tools and Applications*, pp. 1–18, 2022.

[20] Y. Abdalla, M. T. Iqbal, and M. Shehata, "Copy-move forgery detection and localization using a generative adversarial network and convolutional neural-network," *Information*, vol. 10, no. 9, p. 286, 2019, MDPI.

[21] J. H. Bappy, A. K. Roy-Chowdhury, J. Bunk, L. Nataraj, and B. S. Manjunath, "Exploiting spatial structure for localizing manipulated image regions," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4970–4979.

[22] T. Nazir, M. Nawaz, M. Masood, and A. Javed, "Copy move forgery detection and segmentation using improved mask region-based convolution network (RCNN)," *Applied Soft Computing*, vol. 131, p. 109778, 2022, Elsevier.

[23] Dorai, Y., Chausse, F., Gazzah, S., & Amara, N. E. B. (2017). Multi target tracking by linking tracklets with a convolutional neural network. In *VISIGRAPP (6: VISAPP)* (pp. 492–498).

[24] M. Rahim, S. S. Abosuliman, R. Alroobaea, K. Shah, and T. Abdeljawad, "Cosine Similarity and Distance Measures for p, q,–Quasirung Orthopair fuzzy Sets: Applications in Investment Decision-Making," *Heliyon*, vol. 10, no. 1, pp. e02522, 2024.

[25] H. C. Chuang, C. C. Chen, and S. T. Li, "Advancing SVM classification: Parallelizing conjugate gradient for monotonicity enforcement," *Knowledge-Based Systems*, vol. 302, p. 112388, 2024.

[26] I. Amerini, L. Ballan, R. Caldelli, A. Del Bimbo, and G. Serra, "A sift-based forensic method for copy-move attack detection and transformation recovery," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 1099-1110, 2011.

[27] J. Dong, W. Wang, and T. Tan, "Casia image tampering detection evaluation database," in *2013 IEEE China Summit and International Conference on Signal and Information Processing*, 2013, pp. 422–426.

[28] M. N. Abbas, M. S. Ansari, M. N. Asghar, N. Kanwal, T. O'Neill, and B. Lee, "Lightweight deep learning model for detection of copy-move image forgery with post-processed attacks," in *Proc. 19th World Symp. Applied Machine Intelligence and Informatics (SAMI)*, 2021, pp. 1–6.

[29] D. Prabakar, R. Ganesan, D. L. Rani, P. Neti, N. Kalyani, and S. K. Mudradi, "Hybrid deep learning model for copy-move image forgery detection," in *Proc. 2022 Sixth Int. Conf. I-SMAC (IoT Social, Mobile, Analytics Cloud)*, 2022, pp. 1023–1028.

[30] K. Muniappan, R. Patel, S. Kumar, and T. Sharma, "Evaluation of Novel Algorithms in Computational Neuroscience," *Journal of Computational Neuroscience*, vol. 34, no. 2, pp. 211-225, 2023.

[31] S. Booshehrian and E. Amiri, "Copy-move forgery detection and classification using SRVM," in *Proc. 2024 10th Int. Conf. Artificial Intelligence Robotics (QICAR)*, 2024, pp. 359–364.