International Journal of Research in Signal Processing, Computing & Communication System Design
6 (1) 2025, 37-44
www.publishingindia.com/ijrsp

A Comparative Study of Copy-Move Forgery Detection Methods in Digital Images

Muthireddy Rajesh[1], Bokka Sitaram Reddy[2] and Votte Rajashekhar[3]

[1]Assistant Professor, Computer Science and Engineering, Malla Reddy Engineering College for Women, Hyderabad, Telangana, India. Email: muthireddyrajesh747@gmail.com

[2]Assistant Professor, Computer Science and Engineering, Malla Reddy Engineering College for Women, Hyderabad, Telangana, India. Email: ramreddyforyou15@gmail.com

[3]Assistant Professor, Computer Science and Engineering, Malla Reddy Engineering College for Women, Hyderabad, Telangana, India. Email: raj.votte@gmail.com

Abstract: One of the most popular and dishonest image tampering methods is Copy-Move Forgery (CMF), in which a portion of an image is copied and pasted inside another image to hide or change information. It is a common technique for digital manipulation because of its simplicity, which is made possible by widely accessible editing tools. It can be difficult to identify these forgeries, though, particularly if the copied areas undergo post-processing techniques like noise addition, rotation, or scaling. Many algorithms have been created over time to locate and identify CMF; most of them use a similar pipeline, but they vary in how they extract features and match them. Four detection techniques are compared in this paper using images that contain randomly shaped copied regions. The assessment focusses on their efficacy and efficiency, as determined by F-Score, execution time, precision, and recall, offering insights into the advantages and disadvantages of each strategy.

Keywords: Copy-move, DCT, DWT, Feature matching, Image forgery.

I. Introduction

Image forgery has emerged as a substantial issue in multiple fields, including multimedia security, journalism, and scientific literature (Qazi *et al.,* 2013). Image forgery detection techniques are generally classified into two categories: Active and Passive methods (Saranya *et al.,* 2019). Active, or non-blind/intrusive, techniques necessitate the incorporation of supplementary information—such as digital signatures or watermarks—into the original image during its creation, thereby facilitating forgery detection at a subsequent stage. Conversely, passive or blind techniques do not depend on any prior knowledge of the image. They examine the image to detect discrepancies. Passive techniques can reveal alterations including scaling, rotation, blurring, noise introduction, resizing, image splicing, and particularly, copy-move forgery.

Among these, Copy-Move Forgery (CMF) is a prevalent and formidable technique (Redi *et al.,* 2011). This entails replicating a section of an image and repositioning it within the same image to obscure or duplicate objects, as demonstrated in Fig. 1. The duplicated region preserves numerous characteristics of the original image, including texture, colour palette, and noise patterns, rendering the detection of CMF particularly challenging.

Most Copy-Move Forgery Detection (CMFD) algorithms exhibit a similar architecture: they pair highly analogous feature vectors, employing similarity metrics like the Euclidean distance. The efficacy of these algorithms primarily hinges on the selection of feature vectors, which are crucial for identifying tampered areas (Redi *et al.,* 2011). Existing methodologies primarily diverge in two fundamental aspects: the nature of features utilised for matching image blocks and the particular matching strategy implemented.

Pun *et al.* (2015) proposed a method that integrates keypoint and block-based features to enhance detection. Experimental findings indicated that this hybrid methodology surpasses numerous modern CMFD techniques, especially in demanding scenarios including geometric transformations, JPEG compression, and downsampling (Meena and Tyagi, 2019). Wenchang *et al.* (2016) proposed a method that combines Particle Swarm Optimisation (PSO) with SIFT keypoints, utilising the Best Bin First (BBF) algorithm for feature matching. This technique attained a precision of 99%, yet it encountered difficulties in identifying minuscule duplicated regions (Meena and Tyagi, 2019).

This paper presents a comparative analysis of four distinct matching algorithms to examine their influence on the performance of CMFD techniques. The subsequent sections

of the paper are structured as follows: Section II delineates essential components, Section III presents the methodologies, and Sections IV and V encompass the experimental framework, findings, and discourse.

Fig. 1: Example of CMF (CVIP, No Date): Original Image (on the Left); Forged Image (on the Right)

II. Literature Review

In this section, an introduction to the main methods and techniques that are used in the detecting algorithms and also for the comparison is presented. The reader can learn more about those methods by looking up the references listed in each subsection.

Feature extraction and similarity checking parts that match exactly are the critical steps in copy-move forgery detection.

A. Feature Extraction

a) Discrete Cosine Transform (DCT)

The Discrete Cosine Transform (DCT) is a Fourier-related transformation that represents a finite sequence of data points as a sum of cosine functions oscillating at different frequencies. Unlike the Discrete Fourier Transform (DFT), DCT operates using only real numbers. In image processing, transform coding relies on the observation that pixels exhibit a high degree of correlation with their neighboring pixels. The transformation process maps spatially correlated data into uncorrelated coefficients, thereby decorrelating the image data. DCT is widely employed for this purpose, effectively capturing essential image features while reducing redundancy (Yanping et al., 2011).

b) Wavelet Transform Discrete (DWT)

A data vector can be broken down into components of various frequency bands using the Discrete Wavelet Transform (DWT), a linear transform. It effectively captures both spatial and frequency information by working with vectors whose lengths are integer multiples of 2. DWT minimises computational complexity while maintaining crucial image features by concentrating on the most pertinent data (Loai *et al.,* 2017).

c) Scale Invariant Feature Transform (SIFT)

Reliable object or scene matching across various viewpoints is made possible by the Scale Invariant Feature Transform (SIFT), a technique for extracting unique and invariant features from images. These characteristics hold up well against a variety of affine transformations, distortions, shifts in 3D perspective, noise, and changes in illumination. They are also invariant to scale and rotation (Singh *et al.,* 2014).

B. Comparative Analysis

a) K-Means Grouping

Clustering is the process of arranging patterns according to how similar they are. These patterns are usually shown as vectors of measurements or points in a multidimensional space. Compared to patterns in other clusters, patterns within the same cluster are more similar to one another. K-means is one of the most popular clustering methods because of its ease of use and effectiveness. Partitioning n observations into k clusters and allocating each data object to the cluster with the closest mean is the main objective of k-means clustering (Al-Qershi *et al.,* 2016).

b) Accurate Complement

Identical areas within an image are found using the exact match technique. In order to extract matrices and store them in a two-dimensional array, a square block is methodically moved across the image from the top-left to the bottom-right corner. Indistinguishable rows in the array represent identical segments of the image.

Then, rather than being directly compared with other rows, the rows are arranged in lexicographical order. Consequently, rows with comparable pixel values are grouped together, making similarity detection more effective (Arora and Singh, 2019).

c) Study Match

The exact match method is expanded upon by the robust (approximate) match detection technique. It compares their robust representations, which are made up of quantised Discrete Cosine Transform (DCT) coefficients, rather than directly comparing and ranking block representations. According to Friedrich *et al.* (2003), this method increases resistance to small distortions, noise, and compression artefacts.

III. Research Methodology

A. Suggested Enhanced Image Copy-Move Forgery Identification

The suggested algorithm works as follows in the study by Fadl *et al.* (Fadl and Semary, 2014): After being converted to

greyscale, the input image is separated into tiny, overlapping 8 × 8 blocks. The Discrete Cosine Transform (DCT) is used to extract features, and the coefficients that are produced are then rearranged using Zigzag scanning and saved in a matrix. Then, for similarity analysis, the K-Means clustering technique is used. The blocks are saved in a matrix after being clustered and lexicographically sorted. In order to identify forged regions in the image, the correlation between each pair of sorted blocks is finally calculated using the formula given by Fadl and Semary (2014).

B. Image Forgery Detection for High Resolution Images Using SIFT and RANSAC Algorithm

The suggested approach in the study by G. Ramu *et al.* (Ramu and Babu, 2017) is as follows: an input image is read, and the coefficients are obtained by applying the fourth level Discrete Wavelet Transform to it. To create the non-overlapping irregular blocks, the SLIC algorithm applies the computed superpixel. SLIC, or Simple Linear Iterative Clustering, is an algorithm that creates superpixels by assembling pixels according to their colour similarity and proximity in the image plane. A superpixel is a collection of pixels that have similar properties (like pixel intensity). Next, the SIFT algorithm is used to extract the features, and Dot products are calculated to match them.

C. The Copy-Move Forgery Detection Exact Matching Method

The following is the suggested approach in the Jha *et al.* (2020) study: First, the input image is resized to 128 × 128 pixels and converted to greyscale. After that, the Exact Match algorithm is used, which splits the image into overlapping blocks and extracts features that correspond to pixel values from each block. A matrix containing the extracted features is then lexicographically sorted. Ultimately, the matching procedure finds rows in the matrix that are sequentially identical, enabling the corresponding block positions to be determined. Tejas (2018) described how this approach was put into practice.

D. Digital Image Copy-Move Forgery Detection

A technique for identifying copy-move forgeries is suggested in the study by Fridrich *et al*. (2003) and subsequently put into practice by Gaye (2019). The Robust Match technique is used in the procedure. A sliding B×B block is used to scan the input image from the upper-left to the lower-right corner. Each block's coefficients are quantised and saved as a row in a matrix after the Discrete Cosine Transform (DCT) is calculated.

IV. Methodological Framework

A. Preparing the Image Dataset

The Computer Vision and Image Processing Group (CVIP), a division of the University of Palermo's Department of Engineering, provided the dataset used in this investigation (CVIP, no date). Every image in the collection has been resized to 256 × 256 pixels and converted from BMP to JPG format. This particular dataset was selected because, in contrast to many other datasets that contain fixed rectangular, circular, or predefined shapes, the forged regions—copied and pasted segments—occur in a variety of sizes and shapes. The detection process becomes more complex as a result of this variability, more accurately mimicking actual forgery situations.

To establish a feeling of spatial synchronisation and homogeneity with their surrounding areas, the copied regions in the images are positioned in a variety of locations. The dataset is divided into two primary groups: the first group comprises 50 images with a single copied-and-pasted region, and the second group comprises 20 images that have undergone two different kinds of operations: rotation and scaling. The forged areas were rotated at three distinct angles for the rotation operation: 30°, 60°, and 90°. Likewise, the forged regions were subjected to three different scale levels for the scaling operation: 0.75, 1.25, and 1.75.

B. Measures

The following formula is used to determine the average execution time:

Total Execution Time (AvgT) = The quantity of samples

AvgT = Total Execution Time x Number of Samples

The percentage of correctly identified positive instances out of all predicted positives is known as precision (P), and the percentage of correctly identified positive instances out of all actual positives is known as recall (R) (Jason, 2020). The following formula is used to determine these metrics:

P = True Positive + False Positive

Genuinely Positive

R = True Positive

False Negative + True Positive

R = True Positive + False Negative

Genuinely Positive

Precision and Recall are combined into a single metric by the F-Measure (also known as the F1-score) to produce a fair assessment: F-Measure = P+R 2/P·R

V. Result and Analysis

On a system with 12 GB of RAM and an octa-core processor operating at 1.80–1.99 GHz, the experiments were carried out using MATLAB R2017b. In order to simulate copy-move forgeries, the prepared dataset includes a variety of images with various operations applied to the duplicated regions. Every program parameter was maintained at its initial configuration. All rights belong to the respective owners of the source codes used in the experiments, which were retrieved from public repositories.

The outcomes of each approach are examined and contrasted in this section. The precision, recall, F-measure, and execution time for each method are shown in Tables I and II. Figs. 2 and 3 display sample outputs for a representative image to enable a thorough comparison of the algorithms.

Fig. 2: Sample Image Used for the Comparison (for the Unprocessed Forgeries)

Fig. 3: Sample Image Used for the Comparison After Resizing

A. Suggested Enhanced Image Copy-Move Forgery Identification

In comparison to the other approaches, the suggested method took longer to execute. The outcomes, which highlight matched regions in the same colour, show how effective Fast K-Means clustering is. The detection is visually evident in the resultant images, where the localised forged areas, represented by matched points, are displayed as red and blue rectangular blocks.

A sample image for unprocessed forgeries is shown in Fig. 2, and a sample image for processed forgeries is shown in Fig. 3.

Comparing Fig. 4 to the original image, the suggested method did not locate the forged regions as well as it could have, especially for unprocessed copied parts. For the 50 unprocessed copy-move forgery samples, the average execution time was determined to be:

92.79 seconds is the average time.

The algorithm's execution times for processed forgeries were comparable, but as Fig. 5 shows, it struggled to locate the forged areas. Each of the 20 processed samples underwent two operations, rotation and scaling, and the average execution time was determined as follows:

93.65 seconds is the average time.

Notably, the algorithm performed better when identifying square or rectangular cut forgeries in the dataset.

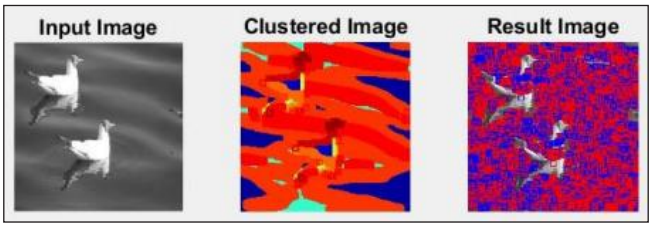

Fig. 4: An Output Sample of the (Fadl and Semary, 2014) Proposed Algorithm

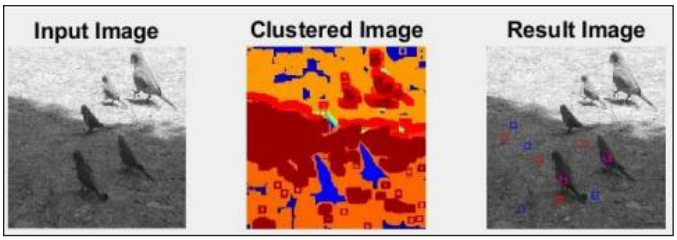

Fig. 5: An Output Sample of the (Fadl and Semary, 2014) Proposed Algorithm. The Sample Forgery is Processed by Scaling it to 1.75

By emphasising the matched blocks on the greyscale image, the suggested method displays the results. This method's adaptable algorithmic structure, which enables future improvements, is one of its benefits. Optimising the correlation and threshold comparison is one possible avenue for future research that could increase the robustness and accuracy of detection.

B. SIFT and RANSAC Algorithms for Detecting Image Forgeries in High-Resolution Images

In addition to the original image, the output displays the DWT result, which displays the HL, LH, and HH, as well as the non-

overlapping irregular blocks and the forgeries displayed in the cluster. A sample output is shown in Fig. 6 of the forgery's heavily replicated areas. AvgT = 7.18 seconds is the average execution time for the 50 unprocessed copy-move forgery samples.

The algorithm performance was relatively the same in terms of execution time for the processed forgeries, and the output was just like the first experiment for pretty much all the samples of the processed forgeries, as shown in Fig. 7. The average execution time for the processed 20 samples (for each sample, two states were taken: rotation and scaling) of copy-move forgeries is:

AvgT = 6.04 seconds.

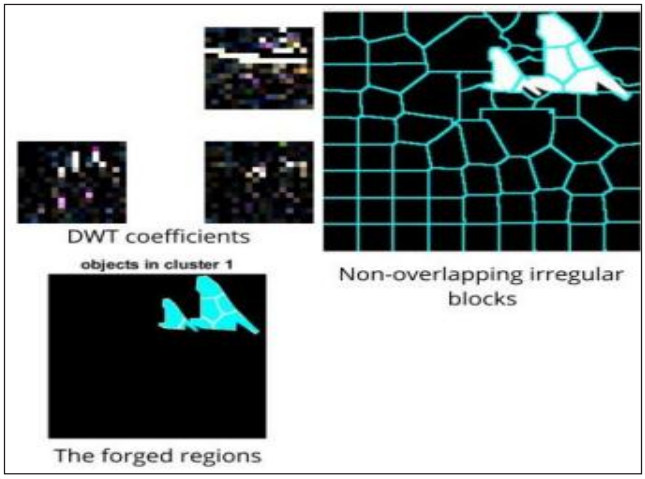

Fig. 6 and 7: An Output Sample of the (Ramu and Babu, 2017) Proposed Algorithm. The Sample Forgery is Processed by Scaling it to 1.75

Comparatively speaking, this method's execution time was quicker than the others. One drawback of the method is that, instead of displaying the predicted regions in the output directly, the algorithm highlights the forged regions based on a mask. The application of the RANSAC algorithm, which successfully lowers unwanted matches, is one of the main benefits. Furthermore, the superpixel size S is calculated using the Haar wavelet, a method that could be modified for other purposes to make the calculation of S simpler.

C. Copy-Move Forgery Detection Using the Exact Matching Method

This method's output displays both the original image and the copied portion. For certain samples, the suggested algorithm was successful in identifying and locating the forgeries by verifying the output by bare eyes to a certain degree with a respectable F-Measure. A sample of the output for a copy-move forgery is displayed in Fig. 8. For the 50 unprocessed copy-move forgery samples, the average execution time is AvgT = 24.06 seconds.

For the processed forgeries, the algorithm's execution times were comparable. Even though it obtained respectable F-Measure values, it was unable to visually identify some of the processed forgeries, just like in the first experiment. A sample output is shown in Fig. 9. The 20 processed samples, each of which was rotated and scaled, had an average execution time of:

23.63 seconds is the average time.

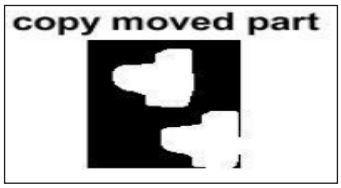

Fig. 8: An Output Sample of the (Jha *et al.,* 2020) Proposed Algorithm

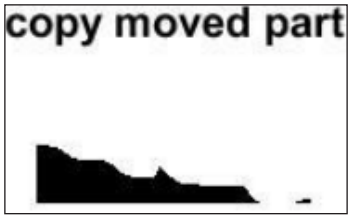

Fig. 9: An Output Sample of the (Jha *et al.,* 2020) Proposed Algorithm. The Sample Forgery is Processed by Scaling it to 1.75

When compared to the original, non-forged images, it is clear that the outputs presented do not accurately localise the forged regions for the majority of samples, despite the research paper's proposal of a Robust Match method. This restriction might be viewed as a methodological flaw. Despite its simplicity, the study shows that the Robust Match method can be used with any copy-move forgery detection algorithm.

Sample results from the algorithms under discussion are shown in Fig. 6–9.

Results from Ramu and Babu (2017) are displayed in Fig. 6 and 7, which demonstrate forgery detection using scaling set to 1.75.

The outputs from Jha *et al.* (2020) are displayed in Fig. 8 and 9, which display the detection results for processed forgeries scaled to 1.75.

D. Identification of Digital Image Copy-Move Forgeries

This method's output shows the F-Measure and the original image along with the forged regions that were found. Fig. 10 displays a sample result. The 50 unprocessed copy-move forgery samples had an average execution time of:

The average time is 22.58 seconds.

The F-Measure for the sample shown in Fig. 10 was determined to be:F-Measure is 0.87.

In comparison to other samples, this value is regarded as good. But in the majority of the 20 processed samples, the suggested method's application was unable to identify forgeries. A sample output of a processed forgery is shown in Fig. 11, where the F-Measure was zero, signifying that the forged regions were completely missed.

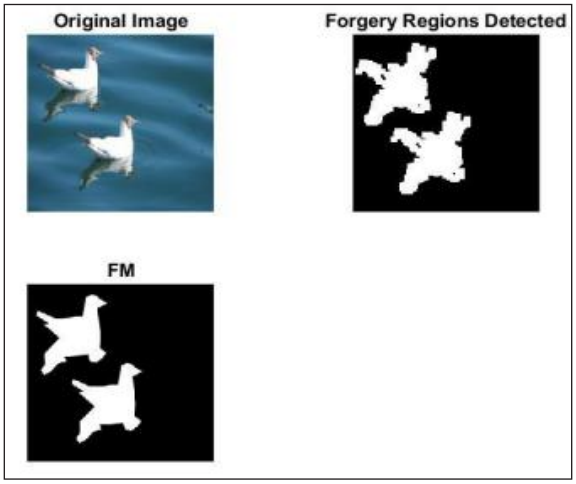

Fig. 10: An Output Sample of the (Fridrich *et al.,* 2003) Proposed Algorithm

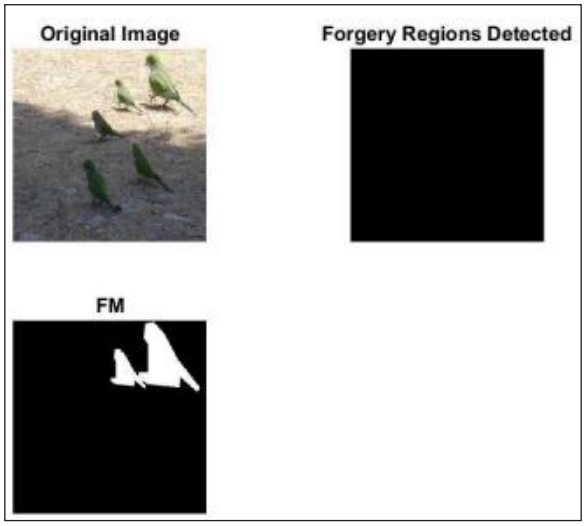

Fig. 11: An Output Sample of the (Fridrich *et al.,* 2003) Proposed Algorithm. The Sample Forgery is Processed by Scaling it to 1.75

A significant drawback of the approach is that while the algorithm does a good job of identifying the majority of plainly copied areas, it is unable to identify forgeries in processed images. This method can be further modified for better performance and is used in the comparison to show technique variation. For instance, as proposed by Fridrich *et al.* (2003), increasing the block size may aid in lowering false matches.

TABLE I

METHOD	BEST METRIC	BEST METRIC VALUE	WORST METRIC	WORST METRIC VALUE
Fadl and Semary (2014)	NA for score metrics	NA	Longest Execution Time	101.13 Seconds
Ramu and Babu (2017)	Fastest Execution Time	7.34 Seconds	Lowest Precision, Recall, F-Score	0.17, 0.14, 0.16
Jha et al. (2020)	Moderate Recall and F-Score	0.42, 0.24	Moderate-High Execution Time	48.96 Seconds
Fridrich et al. (2003)	Highest Precision, Recall, F-Score	0.96, 0.8, 0.87	Moderate Execution Time	22.62 Seconds

Table I, A comparison showing the scores and measures between the four methods based on the sample shown in Fig. 2.

STUDY	EFFECT	OBSERVATION
Ramu and Babu, 2017	Scaling	F-Measure increases slightly
Ramu and Babu, 2017	Rotation	F-Measure decreases
Jha et al., 2020	Rotation	Slight increase in F-Measure
Fridrich et al., 2003	Rotation	Slight increase in F-Measure

Fig. 12 and 13 show how the F-Measure differs as the level of scaling increases and the angle of rotation increases. It can be seen that as the scaling increases, the F-Measure increases a bit accordingly with a slight difference at (Ramu and Babu, 2017). Whilst it appears the F-Measure decreases with the angle of rotation for (Ramu and Babu, 2017), besides slight increases for (Jha *et al.,* 2020) and (Fridrich *et al.,* 2003). The experiments were made on the same sample image of Fig. 3.

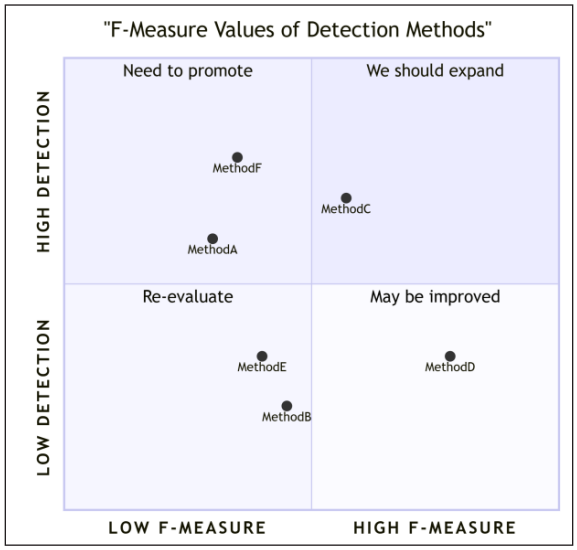

Fig. 12: F-Measure Values of Detection of the Four Methods for the Scaling of 75%, 125%, and 175%

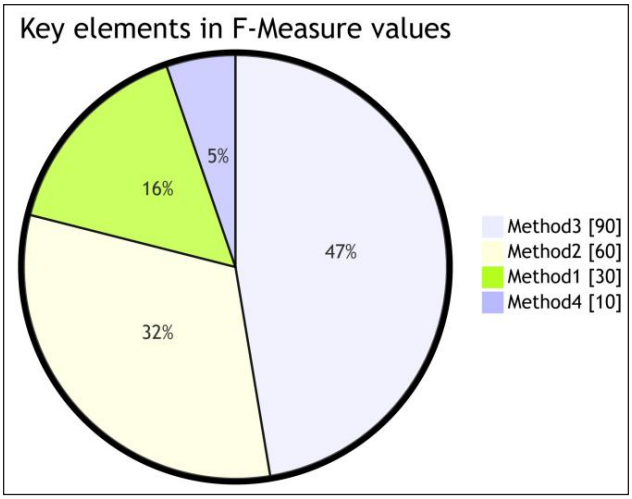

Fig. 13: F-Measure Values of Detection of the Four Methods for the Rotation of Angles 30°, 60°, and 90°

VI. Conclusion

No single approach has been found to be consistently superior to the others, as indicated by a comparative analysis of four techniques for detecting copy-move forgery. Each method involves a compromise among various performance metrics, including accuracy, speed, and robustness.

Frequency-domain block-based techniques, such as those utilizing Discrete Cosine Transform (DCT), demonstrate resilience against noise and compression. However, they may also present challenges due to their computational complexity and reduced effectiveness when dealing with geometric transformations.

On the other hand, keypoint-based techniques are generally more efficient but can struggle in areas characterized by smooth textures.

To improve detection systems, future research could explore the potential of deep learning methodologies, hybrid strategies, and the utilization of larger datasets. This exploration may lead to advancements in the field and enhance the overall effectiveness of forgery detection systems.

In summary, while no single method stands out as the best, the ongoing development and refinement of these techniques are crucial for achieving better performance in copy-move forgery detection. The balance between accuracy, speed, and robustness remains a key consideration for researchers and practitioners alike.

As the field evolves, it is essential to remain open to innovative approaches that may arise, including the integration of artificial intelligence and machine learning into existing frameworks. Such advancements could pave the way for more sophisticated detection systems that are capable of addressing the challenges posed by increasingly complex forgery techniques.

Ultimately, the goal is to create a comprehensive understanding of the strengths and weaknesses of each method, allowing for informed decisions when selecting the appropriate technique for specific applications. By doing so, the effectiveness of copy-move forgery detection can be significantly enhanced, leading to more reliable outcomes in various contexts.

References

[1] M. H. Farhan, K. Shaker, and S. Al-Janabi, "Copy–move forgery detection in digital image forensics: A survey," *Multimedia Tools and Applications*, vol. 83, pp. 70603-70635, 2024.

[2] "Comprehensive survey of block-based, keypoint-based, deep learning based methods; discusses feature extraction, matching, datasets etc.," n.d.

[3] I. A. Ibrahim, M. M. Soliman, K. M. Elsayed, and H. M. Onsi, "Copy move forgery detection techniques: A comprehensive survey of challenges and future directions," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 7, 2021, doi: https://doi.org/10.14569/IJACSA.2021.0120729.

[4] L. Bertojo, C. Néraud, and W. Puech, "A very fast copy-move forgery detection method for 4K ultra HD images," *Frontiers in Signal Processing*, 2022.

[5] "This paper offers a keypoint-based method with a fast matching algorithm (generalized 2NN), aimed at enabling efficient detection even for very large images (4K)," n.d.

[6] E. Amiri *et al.*, "An optimal model for copy-move forgery detection in medical images (CMFMI)," *PMC / NCBI*, 2024.

[7] P. V. Dell'Olmo, "Dataset dependency in CNN-based copy-move forgery detection," *Forensics*, MDPI, 2025.

[8] K. Rehman *et al.,* "Detection of copy-move forgery with deep CNN features," 2025.

[9] B. Benmessahel *et al.*, "Deep learning methods for copy move image forgery detection: A review," *IJSSE*, 2024.

[10] Y. Liu, C. Xia *et al.,* "CMFDFormer: Transformer-based copy-move forgery detection with continual learning," 2023. arXiv preprint.

[11] Y. He, Y. Li, C. Chen, and X. Li, "Image copy-move forgery detection via deep cross-scale PatchMatch," 2023. arXiv.

[12] S. Lu, X. Hu, C. Wang, L. Chen, S. Han, and Y. Han, "Copy-move image forgery detection based on evolving circular domains coverage (ECDC)," 2021.

[13] K. Jha *et al.,* "Digital image forgery detection," *International Research Journal of Engineering and Technology (IRJET),* 2020.

[14] A. Loai *et al.,* "Copy-move forgery detection using integrated DWT and SURF," *Journal of Telecommunication, Electronic and Computer Engineering (JTEC),* 2017, Art. no. 1658.

[15] C. Pun et al., "Image forgery detection: Survey and future directions," *Data, Engineering and Applications*. Singapore: Springer, 2015.

[16] T. Qazi *et al.,* "Oversegmentation and feature point matching," *IEEE Trans. Inf. Forensics Secur.*, vol. 10, pp. 1705-1716, 2013.

[17] J. A. Redi *et al.,* "Survey on blind image forgery detection," *IET Image Process,* vol. 7, pp. 660-670, 2011.

[18] Fadl, and N. Semary, "Digital image forensics: A booklet for beginners," *Multim Tools Appl*., vol. 51, pp. 133-162, 2014.

[19] "A proposed accelerated image copy-move forgery detection," *IEEE Visual Communications and Image Processing Conference*, n.d., pp. 253-257, doi: https://doi.org/10.1109/VCIP.2014.7051552.