

A deep learning model with an inductive transfer learning for forgery image detection

Prabhu Bevinamarad¹, Prakash H. Unki², Venkatesh Bhandage³

¹Department of Computer Science and Engineering, BLDEA's V.P. Dr. P.G. Halakatti College of Engineering and Technology (Affiliated to Visvesveraya Technological UniversityBelagavi), Vijayapur, India

²Department of Information Science and Engineering, BLDEA's V.P. Dr. P.G. Halakatti College of Engineering and Technology (Affiliated to Visvesveraya Technological UniversityBelagavi), Vijayapur, India

³Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal 576104, Karnataka, India

Article Info

Article history:

Received Apr 16, 2024

Revised Sep 25, 2024

Accepted Sep 30, 2024

Keywords:

Copy-move forgery

Deep learning

Image tampering

Image-splicing

Inductive transfer learning

ABSTRACT

Due to the availability of affordable electronic devices and several advanced on-line and offline multimedia content editing applications, the frequency of image manipulation has increased. In addition, the manipulated images are presented as evidence in courtrooms, circulated on social media and uploaded upon authentication to deceive the situation. This study implements a deep learning (DL) framework with inductive transfer learning (ITL) by using a pre-trained network to benefit from the discovered feature maps rather than starting from scratch and fine-tuning the process to check and classify whether the suspected image is authenticated or forged effectively. To experiment with the proposed model, we used both Columbian uncompressed image splicing detection (CUISD) and the CoMoFoD dataset for training and testing. We measured the model's performance by changing hyperparameters and confirmed the better selection of values for the hyperparameter to yield compromised results. As per the evaluation results, our model showed improved results by classifying new instances of images with an average precision of 89.00%, recall of 86.43%, F1-score of 87.32, and accuracy of 87.72% and consistently performed better compared to other methods currently in use.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Venkatesh Bhandage

Department of Computer Science and Engineering, Manipal Institute of Technology

Manipal Academy of Higher Education

Manipal 576104, Karnataka, India

Email: venkatesh.bhandage@manipal.edu

1. INTRODUCTION

Today, multimedia content has become a significant part of various automated digital systems, such as user authentication, and online user document verification. Multimedia defines digital content that includes audio, video, images, and combinations. Combining these automated digital systems and various multimedia data has created state-of-the-art technologies that provide various digital services to humankind to replace traditional methods and increase productivity. On the other hand, the increased sophistication of image editing tools, forgery applications, and artificial intelligence (AI) algorithms caused a swift upsurge in digitally manipulated counterfeits in internet crime worldwide [1]. For instance, criminals carry out criminal activities and pose a significant threat to internet users by tampering with images and videos. They create altered images

to misrepresent their meaning with malicious intentions and fake evidence or documents to present in courtrooms without significantly leaving visual clues to be detected by the naked eye. There are many different ways to manipulate multimedia content, and such instances of digitally fabricated photos from the past have been preserved in Farid's collection of web pages [2]. Think of copy-move image forgery as one of the forgery examples; in this case, some copied parts are placed at another location within the same image. The two image sources are combined to form a forgery image in image splicing. Similar tampering is also performed in the case of audio and video data [3]–[5]. In all these types of multimedia tampering, image tampering has gained much attention due to the availability of cameras with excellent resolution, state-of-the-art photo-editing applications, and advanced software tools.

The first block-based forgery detection approach was introduced in [6]. Since then, various image tampering detection techniques have been developed, incorporating diverse feature extraction and matching methods. With the increasing popularity of deep neural network frameworks, there have been efforts to utilize convolutional neural networks (CNNs) to enhance forgery image detection. This section summarizes the latest image tampering detection approaches in recent years.

Bayar and Stamm [7] introduced a CNN with layers dedicated to suppressing the content of an image and learning manipulation detection features adaptively. Ouyang *et al.* [8] employs a pre-trained model already created from an extensive database, such as ImageNet. Then it modifies the net structure slightly using tiny training copy-move samples to find copy-move image samples generated automatically by computer. Huang *et al.* [9] describes a CNN that can understand features extracted from each convolutional layer and autonomously learn features to recognize different types of image manipulation. Muzaffer and Ulutas [10] discusses the detection and localization of copy-move forgeries using a DL-based framework instead of conventional feature extraction methods. Abdalla *et al.* [11] suggested a CNN model with added pre-processing layers to detect different copy-move forgery images. The experiments demonstrate that the total validation accuracy stands at 90%. Elaskily *et al.* [12], mention CNN trains hierarchical features represented from an input image to identify altered and original images. Rodriguez-Ortega *et al.* [13], proposed two approaches, a model using a custom architecture and a model using transfer learning to distinguish between altered and original images. Abbas *et al.* [14] proposed two lightweight SmallerVGGNet (inspired by VGGNet) and MobileNetV2 deep learning (DL) models to classify and detect copy-move forgery and post-forgery images. Abhishek and Jindal [15] proposes a method based on color illumination, deep CNN, and semantic segmentation to detect and localize image forgeries. Goel *et al.* [16] introduces a DL approach with a dual-branch CNN to detect passive copy-move forgery by extracting multi-scale features using various kernel sizes. Kadam *et al.* [17] presents a lightweight model constructed using mask regional CNN (R-CNN) with MobileNet V1 to detect the forgeries present in an image along with corresponding percentages. Fahn and Wu [18], proposed a DL-based method for detecting forgery images. This method utilizes discrete fourier transform and contrastive learning, enabling the model to directly learn the differences between authentic and forged images. Mehrjardi *et al.* [19] presents a DL method for image-level forgery detection by employing a pre-trained deep model and global average pooling (GAP). Additionally, pixel-level forgery detection is achieved through heatmap activation. Similar kinds of DL models with transfer learning and particle swarm optimization (PSO) techniques are discussed in [20]–[22], respectively. Sadanand *et al.* [23] proposed CNN with error level analysis error level analysis (ELA) adopted to detect and accurately classify the copy-move forgery images. He *et al.* [24], proposed a method for detecting GAN-generated forgery images by combining central difference convolution and vanilla convolution (CDC-Mix). This approach considers the depth and width features of neural networks and analyzes the impact of attention on network performance.

Although many DL models mentioned above have demonstrated promising results in classifying images as forged or authentic, they have some drawbacks, including the need for improved forgery image detection frameworks, generalization to unseen forgeries, and the possibility that training DL models for copy-move forgery detection will take a long time and require powerful hardware. Therefore, this paper presents a DL-based model using inductive transfer learning and fine-tuning processes to improve the model's ability to detect altered images and reduce loss or misclassification. The model is trained using high-quality authentic and forgery images with augmented forms to enhance the learning ability. The proposed work also presents a better selection of significant hyperparameter values to enhance the training and testing performance and attain a model generalization. Hence, our model can be integrated with various web applications, government agencies, and social media platforms as a backend to verify the legitimacy of image data and ensure that forged images are not circulated on social media.

2. PROPOSED METHOD

The primary focus of our proposed methodology is to implement a reliable custom DL model using the pre-trained network and apply the inductive transfer learning and fine-tuning process to extract relevant and discriminative image characteristics that are well-structured and acceptable for the network to learn and produce better classification results. Therefore, we used the MobileNet V2 model, developed at Google, and pre-trained on the ImageNet dataset, which consisted of 1.4M images and 1,000 classes. We adopted inductive transfer learning to take benefit of discovered feature maps rather than starting them from scratch. On the other hand, the fine-tuning process enables the higher-order feature representations of the base model to make them more pertinent to the particular classification task and efficiently classify the new sample images as forgery or authenticated. Transfer learning and fine-tuning add a significant step to our proposed framework. In inductive transfer learning, we frame a new fully connected layer on top of the pre-trained model and a classifier to use the representations learned by a base network and extract significant features from a new set of query image samples. On the other hand, the fine-tuning process unfreezes some of a frozen model base’s top layers and jointly trains the base model’s final layers and newly added classifier layer to “fine-tune” the higher-order feature representations of the base model to prevent any modifications to the weights of base layers during the backpropagation phenomenon. Figure 1 depicts the schematic design of the suggested model.

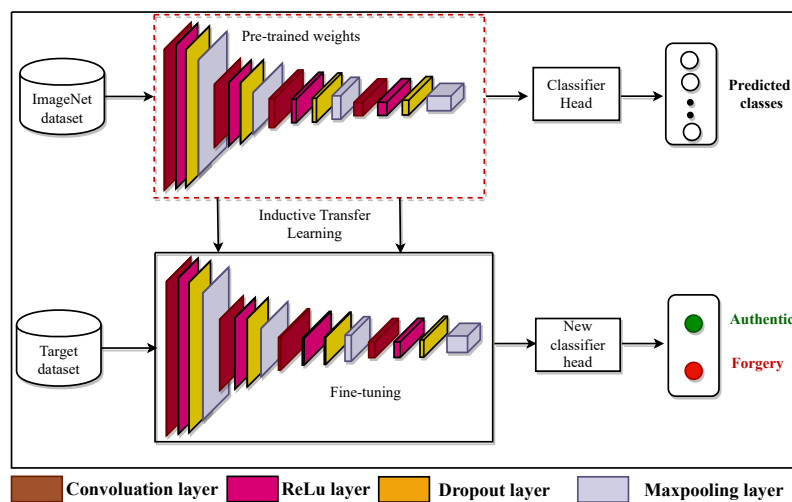








Figure 1. Schematic design of our proposed DL model with inductive transfer learning

2.1. Description of target dataset

To evaluate the goals of the suggested model, we have considered two popularly known and publicly accessed datasets, i.e., CoMoFoD [25] and Columbia uncompressed image splicing detection (CUIISD) [26]. The former primarily contains 200 copy-move forgery images of size 512×512 and uses visual appearance-related post-processing attacks. As a result, 10,400 image samples, including original, forgery and ground truth images in the CoMoFoD dataset. The latter includes 363 total image samples in the entire dataset. Out of them, spliced images are 180 and 183 are authentic. The dimension of each image ranges from 757×568 to 1152×768 . Table 1 depicts authentic and forged image samples from two targeted datasets.

Table 1. Authentic and forged image samples from target datasets

CUIISD dataset [26]			CoMoFoD [25]		
					
Authentic	Forged	Forged	Authentic	Forged	Forged

2.2. Implementation details

The proposed methodology uses Python 3.8, Keras and TensorFlow as a backend toolkit to incorporate inductive transfer learning and fine-tuning process and replace the prior pre-trained FC layer with a new FC layer that is framed as a model head and attached to the base of the target model. Later, the framework's primary layers are frozen to avoid weight updates during backpropagation. Subsequently, a dense layer with adequate classification outputs and a softMax function is added to the global average pooling layer to create a proper framework for accurately classifying the findings. Since our model intends to identify only two classes (forgery or authentic), two outputs and SoftMax function are added to the dense layer. The model can handle input images up to $300 \times 300 \times 3$. During preprocessing, the selected image size is reduced based on the model input designed (300×300 pixels) and image pixel values scaled between 0 and 1 range. Later, the training dataset size increased artificially by using the data augmentation technique to create altered versions of the images to extract significant image features from all perspectives to improve learning and classification ability. The suggested approach uses Keras's DataGenerator class to enhance image data. In this step, every batch of training data is randomly rotated, cropped, and resized. The implementation details in terms of pseudo-code are presented in Algorithm 1.

Algorithm 1. Proposed forgery detection methodology

Require: Images from the dataset: D
Ensure: The input image is Forgery or Authentic

- 1: **function** RESIZE($D, [h, w, d]$)
- 2: **end function**
- 3: **function** SPLIT(D, ratio)
- 4: **end function**
- 5: **function** NORMALIZE($images, [0, 1]$)
- 6: **end function**
- 7: Load the target domain image dataset.
- 8: $D \leftarrow$ Target domain image dataset.
- 9: Preprocess data from both domains:
- 10: $images \leftarrow$ RESIZE($D, [300 \times 300 \times 3]$)
- 11: NORMALIZE($images, [0, 1]$)
- 12: Perform data augmentation such as random rotations, flips, and shifts.
- 13: Split D : Divide 80% for training and 20% for validation.
- 14: $[D_T, D_V] \leftarrow$ SPLIT($D, 0.2$)
- 15: **function** MODEL(D_T, D_V)
- 16: Load the MobileNet V2 model and modify it for inductive transfer learning
- 17: **function** LOAD_MOBILENET_V2($input_shape, num_classes, weights = imagenet$)
- 18: Set the base model's layers in a frozen state.
- 19: $base_model.trainable \leftarrow False$
- 20: Add custom top layers for transfer learning
- 21: $x \leftarrow base_model.output$
- 22: $x \leftarrow$ GLOBALAVERAGEPOOLING2D
- 23: $x \leftarrow$ DENSE(128, $activation = 'relu'$)
- 24: $Y_{pred} \leftarrow$ DENSE($num_classes, activation = 'softmax'$)
- 25: Create transfer learning model
- 26: $model \leftarrow$ MODEL($inputs=base_model.input, outputs=Y_{pred}$)
- 27: **end function**
- 28: **end function**
- 29: $Model \leftarrow$ MODEL(D_T, D_V)
- 30: Compile the modified MobileNetV2 model
- 31: $result \leftarrow$ COMPILE_TRAIN($Model, optimizer = adam, loss = binary_cross_entropy$)
- 32: Evaluate the model and compute classification metrics.

2.3. Model training and testing

The training and testing experiments are simulated using 290 and 400 (including original and forgery) images from both datasets. Some authentic images from the CoMoFoD dataset are added to the CUISD dataset to increase the number of authentic images during training sessions. The dataset is divided into 80:20 ratios for training and testing. A predetermined set of initial hyperparameter values is kept constant during initial experimentation to assess the training and testing outcome. The parameters include input image size of 300×300 , initial weight set to "ImageNet," 40 epochs, and batch size=32 with early stopping based on minimum validation loss with the patience of 10. The optimizer is set to "Adam," with a learning rate of 0.000001.

Later, during the simulation, experiments are repeated by setting a different value for the epoch, batch size and learning rate to identify the significant value at which the proposed DL model achieves a better classification result. Figure 2 depicts the training and validation accuracy and the loss curves obtained during the training and validation phase.

As per the graphs in Figures 2(a) and 2(b) for training and validation over 40 epochs. The accuracy of the training and validation varies from 0.78 to 0.92. The training accuracy starts around 0.85. Exhibits significant fluctuations throughout the epochs. The accuracy does not show a clear increasing trend and varies between 0.80 and 0.88. The validation accuracy starts high, around 0.92, but drops slowly in the first few epochs to below 0.80. After the initial drop, it stabilizes and fluctuates around 0.88 to 0.90. It appears relatively stable but with noticeable fluctuations compared to the accuracy of the training. On the other hand, the training and validation loss for epochs 0 to 40 varies between a loss of 0.22 and 0.38. The training loss starts around 0.35 and shows a general decreasing trend with some fluctuations. The loss decreases to approximately 0.25 by the end of the 40 epochs. The validation loss starts high, around 0.38, but decreases rapidly in the first few epochs to around 0.30 and continues to decrease gradually with some fluctuations and shows improvement overall, reaching approximately 0.22 by the end of the 40 epochs.

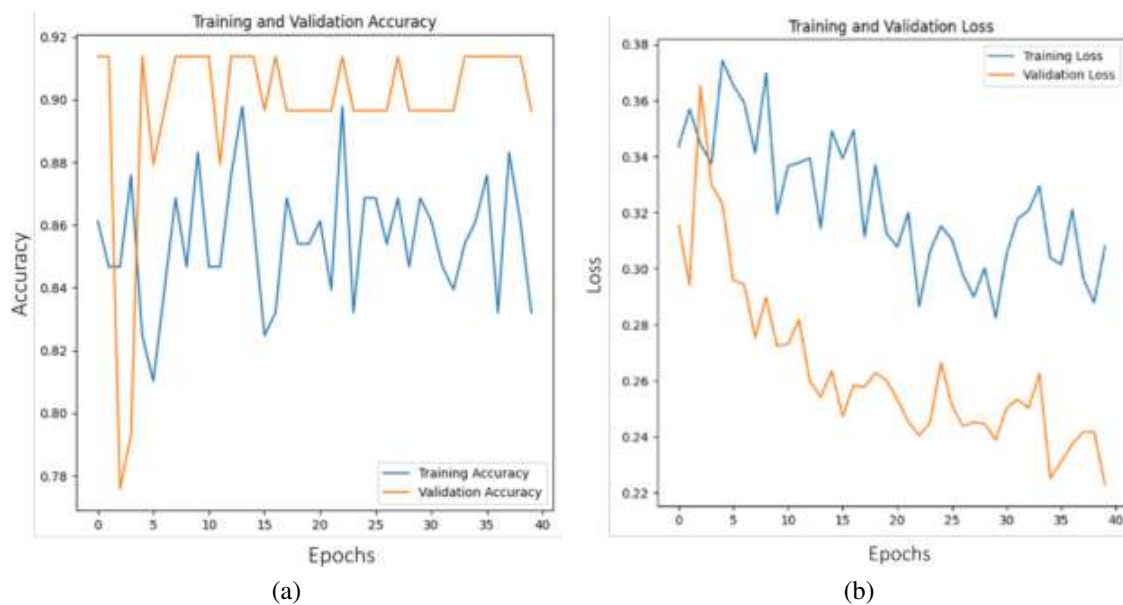


Figure 2. Accuracy and loss curve (a) training and validation accuracy and (b) training and validation loss

In summary, the training accuracy shows high variability and gradually exhibits an upward trend, indicating little inconsistent learning. The validation accuracy starts high, drops initially, and stabilizes with fluctuations. The fluctuations in accuracy suggest potential issues with the model's stability or overfitting. The training loss decreases over time, indicating that the model is learning. The validation loss decreases overall, suggesting that training improves the model's performance on unseen data.

3. MODEL EVALUATION AND RESULT DISCUSSION

The experimental trials were conducted on an Intel core i7 5.1GHz 64-bit processor with 16 GB RAM and a 4 GB GPU to train and test the model. This section includes a description of evaluation metrics, evaluation of the suggested model and a discussion of the evaluation results.

3.1. Evaluation metrics

To understand the effectiveness of our model, we have adopted evaluation metrics precision (P), recall (R), harmonic mean (F1-score), and accuracy (Acc). The (1), (2), (3), and (4) define the computation of these evaluation metrics.

$$P = \frac{(T^+)}{(T^+ + F^+)} \times 100 \quad (1)$$

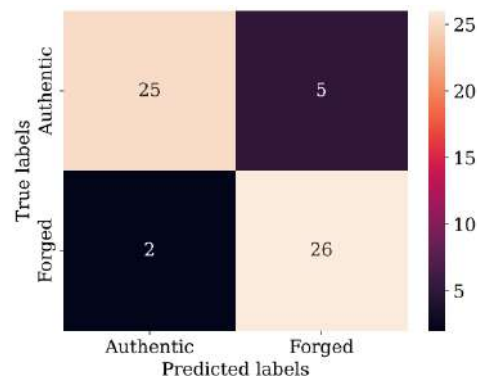
$$R = \frac{(T^+)}{(T^+ + F^-)} \times 100 \quad (2)$$

$$F1 = 2 \times \frac{P \times R}{(P + R)} \times 100 \quad (3)$$

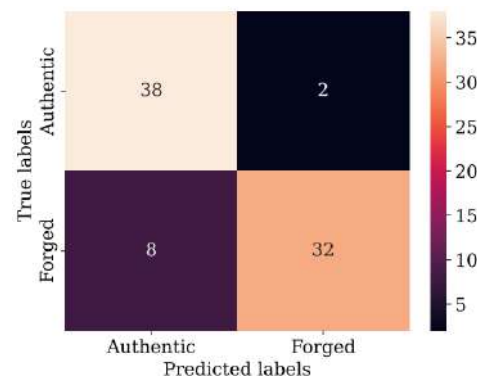
$$Acc = \frac{(T^+ + T^-)}{(T^+ + F^- + T^+ + F^-)} \times 100 \quad (4)$$

Where, true positive (T^+) relates to how many forgery images are correctly classified as a forgery, false positive (F^+) refers to how many authentic images are inaccurately discovered as forged images, true negatives (T^-) refers to how many authentic images are accurately determined as authentic, and false negative (F^-) refers to incorrectly identifying the count of forgery images as authentic.

Later, the adequately trained model is tested with 58 unseen forged images and 80 unseen authentic images from both datasets. Classification results are presented in the confusion matrix depicted in Figure 3 for CUIISD shown in Figure 3(a) and CoMoFoD shown in Figure 3(b) dataset respectively. Performance metrics computed using TP, TN, FP, and FN yield the following results: for CoMoFoD, precision is 83.87%, recall is 92.86%, F1-score is 88.14%, and accuracy is 87.93%; for the Columbia Uncompressed Image Splicing dataset, precision is 94.12%, recall is 80%, F1-score is 86.49%, and accuracy is 87.5%. The detected results showing the actual label, the predicted label, and the image name are detailed in Table 2 for CUIISD and CoMoFoD dataset respectively.



(a)



(b)

Figure 3. Confusion matrix for test sets from (a) CUIISD and (b) CoMoFoD dataset

3.2. Discussion of results over target dataset

The observed results are compared with other state-of-the-art reference models to assess how well our model works. Table 3 provides a detailed summary of reference model specifications and corresponding accuracies, and Figure 4 illustrates the performance comparison in terms of precision shown in Figure 4(a), recall shown in Figure 4(b), F1-score shown in Figure 4(c), and accuracy shown in Figure 4(d). As per the results, the dual branch CNN [16] achieves the highest performance with an F1 score of 94 and a perfect recall, making it ideal for tasks where missing any positive instance is unacceptable. Our model, DL with ITL stands out for its high precision of 89.00 and competitive F1 score of 87.32, indicating its effectiveness in scenarios where precision is as important as recall. MobileNet V1 [17] shows the lowest F1 score at 64.20, suggesting it is less effective than other models. MobileNet V2 [14] performs better, with an F1 score of 84.40, indicating improvements over V1. CNN variants [11] and [20] show varied performance, with [11] achieving an F1 score of 88.35 and [20] scoring 82.00, reflecting recent improvements in CNN design and optimization. The CNN with ELA [23] performs well, with an F1 score of 85.90 and the second-highest precision of 88.10, showcasing advancements in ELA techniques. The performance metric F1 score balances precision and recall. The dual branch CNN [16] achieved the highest F1 score of 94, indicating superior overall performance. On the other hand, our model, with an F1 score of 87.32, is competitive and higher than most other models except [11] and [16] and achieved the highest precision of 89.00, indicating fewer false positives. Hence, our model demonstrates strong performance across all metrics, excelling in precision and competitive recall, making it a robust choice for image forensic applications.

Table 2. Visualization of detection results obtained for CUISD and CoMoFoD dataset









CUISD dataset		CoMoFoD dataset	
canong3_kodakdcs330_sub.07	canong3_kodakdcs330_sub.09	003_F_CA2	001_F_CA2
			
Actual: Forged canong3_02_sub.01	Actual: Forged canong3_02_sub.02	Actual: Forged 002_F_JC2	Actual: Forged 002_O_BC2
			
Actual: Authentic	Actual: Authentic	Actual: Forged	Actual: Authentic

Table 3. Comparison of model specification and accuracy with reference models

Reference models	[11]	[17]	[14]	[16]	[20]	[23]	Ours
Model name	CNN	MobilNet V1	MobilNet V2	Dual branch CNN	CNN	CNN with ELA CASIA V1.0,	DL with ITL CoMoFoD
Dataset used	Composite	Composite	Composite	MICC- F2000	Composite	CASIA V2.0 and MICC	and CUISD
Input image size	64×64	512×512	224×224	700×700	128×128	128×128	300×300
Training set size	1254	2505	2260	1700	2232	3772	552
Testing set size	537	318	565	150	1488	943	138
No. of epoch	7 with 90 iterations	240	100	100	30	30	40
Overall accuracy	90%	NA	85.6%	96%	87%	81.1%	87.72%

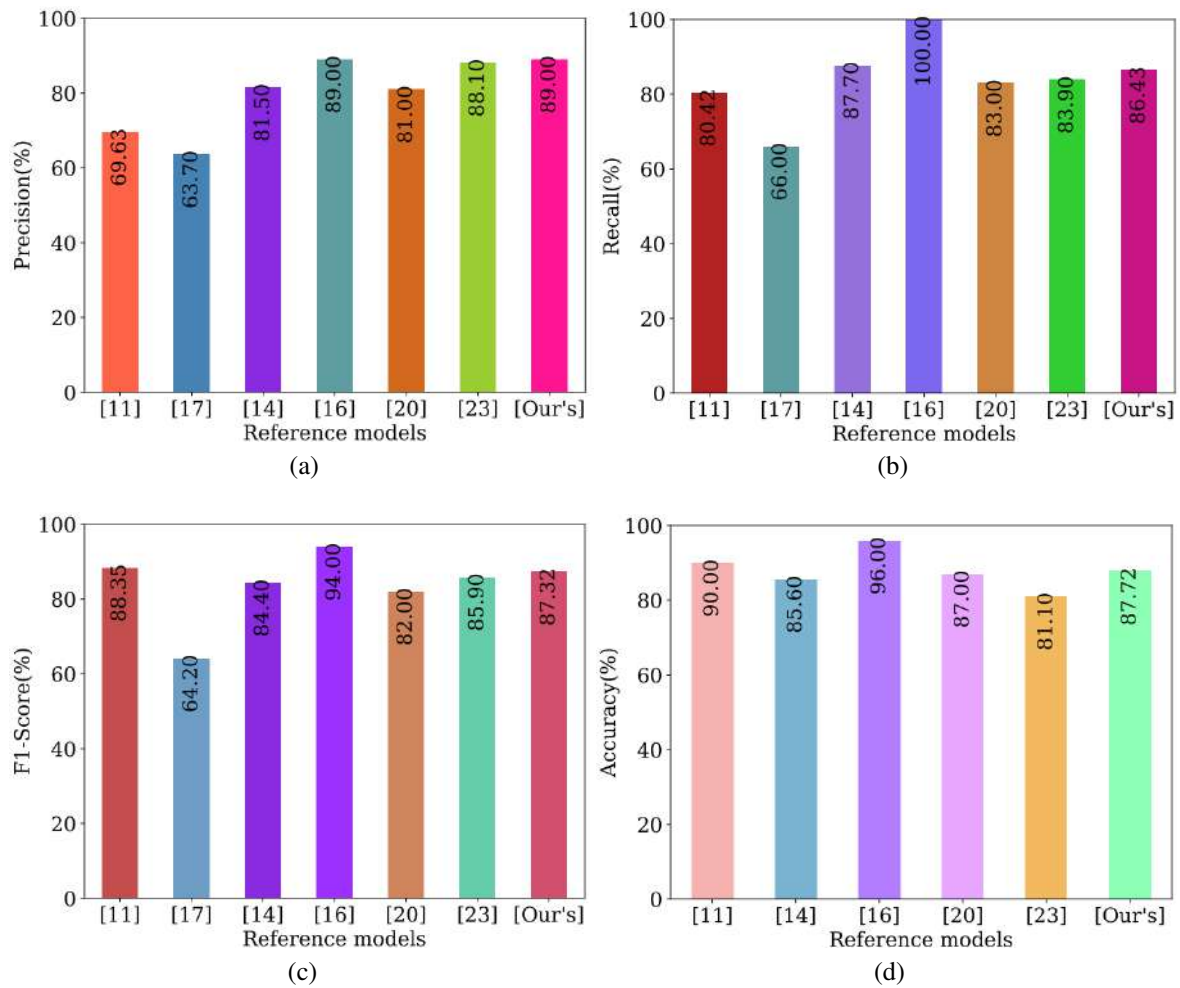


Figure 4. Comparison of performance metrics: (a) precision, (b) recall, (c) F1-score, and (d) accuracy with reference models





4. CONCLUSION

Computer vision-based image authentication has many cutting-edge applications across various fields, including banking, forensics, and medicine. The proposed methodology has tried to deliver the system to accurately classify the authentic and forgery images for both CUISD and CoMoFoD datasets. Our model implements a DL concept with inductive transfer learning and a fine-tuning mechanism to efficiently utilize the feature maps from the pre-trained model and classify forgery images. According to the evaluation's findings, the suggested model yields an average detection accuracy of around 87.72% and precision of 89.00%, recall of 86.43%, and F1-score of 87.32. The comparative study in the results and discussion section shows that it performs better than other models. Currently, the proposed model can only confirm image authenticity without marking the exact location of the tampering present in an image. In addition, the proposed DL model requires extensive input data for the training with an enhanced system consisting of GPU configuration to generate good results. Usually, a system with inadequate resources takes more time and produces poor classification results. Hence, considering these two issues mentioned above, the present work would be extended by training large and diverse image datasets to increase recall and other performance metrics and highlight the forgery image regions.





REFERENCES

- [1] A. Chadha, V. Kumar, S. Kashyap, and M. Gupta, "Deepfake: an overview," *Proceedings of second international conference on computing, communications, and cyber-security: IC4S 2020*, pp. 557–566, 2021, doi: 10.1007/978-981-16-0733-2_39.
- [2] R. Kumari and H. Garg, "An image copy-move forgery detection based on SURF and fourier-mellin transforms," *2023 International Conference on Artificial Intelligence and Smart Communication, AISC 2023*, pp. 515–519, 2023, doi: 10.1109/AISC56616.2023.10085429.
- [3] P. R. Bevinamarad and M. S. Shirdonkar, "Audio forgery detection techniques: present and past review," in *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, Jun. 2020, pp. 613–618, doi: 10.1109/ICOEI48184.2020.9143014.
- [4] P. R. Bevinamarad and M. U. Mulla, "Review of techniques for the detection of passive video forgeries," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 2, pp. 199–203, 2017.
- [5] N. A. Shelke and S. S. Kasana, "Multiple forgery detection in digital video with VGG-16-based deep neural network and KPCA," *Multimedia Tools and Applications*, vol. 83, no. 2, pp. 5415–5435, Jan. 2024, doi: 10.1007/s11042-023-15561-0.
- [6] J. Fridrich, D. Soukal, and J. Lukas, "Detection of copy-move forgery in digital images," in *Proceedings of Digital Forensic Research Workshop*, 2003, vol. 3, no. 2, pp. 652–663.
- [7] B. Bayar and M. C. Stamm, "A deep learning approach to universal image manipulation detection using a new convolutional layer," in *Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security*, Jun. 2016, pp. 5–10, doi: 10.1145/2909827.2930786.
- [8] J. Ouyang, Y. Liu, and M. Liao, "Copy-move forgery detection based on deep learning," in *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, Oct. 2017, pp. 1–5, doi: 10.1109/CISP-BMEI.2017.8301940.
- [9] N. Huang, J. He, and N. Zhu, "A novel method for detecting image forgery based on convolutional neural network," in *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, Aug. 2018, pp. 1702–1705, doi: 10.1109/Trust-Com/BigDataSE.2018.00255.
- [10] G. Muzaffer and G. Ulutas, "A new deep learning-based method to detection of copy-move forgery in digital images," in *2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT)*, Apr. 2019, pp. 1–4, doi: 10.1109/EBBT.2019.8741657.
- [11] Y. Abdalla, M. Iqbal, and M. Shehata, "Convolutional neural network for copy-move forgery detection," *Symmetry*, vol. 11, no. 10, p. 1280, Oct. 2019, doi: 10.3390/sym11101280.
- [12] M. A. Elaskily *et al.*, "A novel deep learning framework for copy-move forgery detection in images," *Multimedia Tools and Applications*, vol. 79, no. 27–28, pp. 19167–19192, Jul. 2020, doi: 10.1007/s11042-020-08751-7.
- [13] Y. Rodriguez-Ortega, D. M. Ballesteros, and D. Renza, "Copy-move forgery detection (CMFD) using deep learning for image and video forensics," *Journal of Imaging*, vol. 7, no. 3, p. 59, Mar. 2021, doi: 10.3390/jimaging7030059.
- [14] 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 *2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI)*, Jan. 2021, pp. 125–130, doi: 10.1109/SAMI50585.2021.9378690.
- [15] Abhishek and N. Jindal, "Copy move and splicing forgery detection using deep convolution neural network, and semantic segmentation," *Multimedia Tools and Applications*, vol. 80, no. 3, pp. 3571–3599, Jan. 2021, doi: 10.1007/s11042-020-09816-3.
- [16] N. Goel, S. Kaur, and R. Bala, "Dual branch convolutional neural network for copy move forgery detection," *IET Image Processing*, vol. 15, no. 3, pp. 656–665, Feb. 2021, doi: 10.1049/ipr2.12051.
- [17] K. D. Kadam, S. Ahirrao, K. Kotecha, and S. Sahu, "Detection and localization of multiple image splicing using MobileNet V1," *IEEE Access*, vol. 9, pp. 162499–162519, 2021, doi: 10.1109/ACCESS.2021.3130342.
- [18] C.-S. Fahn and T.-C. Wu, "A deep-neural-network-based approach to detecting forgery images generated from various generative adversarial networks," in *2022 International Conference on Machine Learning and Cybernetics (ICMLC)*, Sep. 2022, pp. 115–123, doi: 10.1109/ICMLC56445.2022.9941295.
- [19] F. Z. Mehrjardi, A. M. Latif, and M. S. Zarchi, "Copy-move forgery detection and localization using deep learning," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 37, no. 09, p. 2352012, Jul. 2023, doi: 10.1142/S0218001423520122.
- [20] T. Muniappan, N. B. A. Warif, A. Ismail, and N. A. M. Abir, "An evaluation of convolutional neural network (CNN) model for copy-move and splicing forgery detection," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 11, no. 2, pp. 730–740, 2023.
- [21] A. H. Khalil, A. Z. Ghalwash, H. A.-G. Elsayed, G. I. Salama, and H. A. Ghalwash, "Enhancing digital image forgery detection using transfer learning," *IEEE Access*, vol. 11, pp. 91583–91594, 2023, doi: 10.1109/ACCESS.2023.3307357.
- [22] Ö. Kasim, "Deep learning-based efficient and robust image forgery detection," *Multimedia Tools and Applications*, vol. 83, no. 21, pp. 59819–59838, Jan. 2024, doi: 10.1007/s11042-023-17946-7.
- [23] V. S. Sadanand, S. S. Janardhana, S. Purushothaman, S. Hande, and R. Prakash, "Convolutional neural network-based techniques and error level analysis for image tamper detection," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 33, no. 2, pp. 1100–1107, Feb. 2024, doi: 10.11591/ijeecs.v33.i2.pp1100-1107.
- [24] D. He *et al.*, "MCDC-Net: multi-scale forgery image detection network based on central difference convolution," *IET Image Processing*, vol. 18, no. 1, pp. 1–12, Jan. 2024, doi: 10.1049/ipr2.12928.
- [25] D. Tralic, I. Zupancic, S. Grgic, and M. Grgic, "CoMoFoD - new database for copy-move forgery detection," *Proceedings Elmar - International Symposium Electronics in Marine*, pp. 49–54, 2013.
- [26] Y. Hsu and S. Chang, "Detecting image splicing using geometry invariants and camera characteristics consistency," in *2006 IEEE International Conference on Multimedia and Expo*, Jul. 2006, pp. 549–552, doi: 10.1109/ICME.2006.262447.





BIOGRAPHIES OF AUTHORS

Prabhu Bevinamarad     completed his B.E. in Information Science and Engineering and M.Tech. Computer Science and Engineering from Visvesvaraya Technological University, Belagavi, Karnataka, India, in 2008 and 2013, respectively. He is pursuing a Ph.D. in Computer Science and Engineering from Visvesvaraya Technological University, Belagavi, Karnataka, India. His research interests include deep learning, digital images, and audio forensics. He can be contacted at email: prabhubev@gmail.com.



Prakash H. Unki     completed his B.E. in Electronics and Communication from Karnataka University, Dharwad and M.Tech. and Ph.D. in Computer Science and Information Sciences from Visvesvaraya Technological University, Belagavi, Karnataka, India. He is working as a Professor and Head in the Department of Information Science and Engineering, B.L.D.E.A's V. P. Dr. P. G. H. College of Engineering and Technology, Vijayapur. He has 26 years of teaching experience in an engineering college. He published 21 research articles in various international journals and conferences. Currently, he is guiding two doctoral students. His research interests include digital image processing, pattern recognition, machine learning, cloud computing, data science, and big data. He is a member of various professional societies. He can be contacted at email: prakashunki@gmail.com.



Venkatesh Bhandage     currently working as an Assistant Professor (Senior Scale) in the Department of Computer Science and Engineering at the Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal. He completed his Ph.D. from Visvesvaraya Technological University, Belagavi, in 2020. His research interests include image processing, artificial intelligence, medical image processing, and machine learning. He has published 12 papers in international journals and conferences. Currently, he is guiding two research scholars. Additionally, he serves as a reviewer for several journals, including Springer Nature Computer Science (SNCS), PLOS ONE, Scientific Reports, and Signal, Image, and Video Processing. He is a member of IEEE, IE, and ISTE. He can be contacted at email: venkatesh.bhandage@manipal.edu.