WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal Homepage: www.ijrpr.com ISSN 2582-7421

Synthetic Media Tool Detection

Santosh Kumar∗, Preeti R G†, Priyanka S C†, Rakshitha S B†, Shravya B H†

∗Assistant Professor, Department of Computer Science and Engineering
†UG Scholars, Department of Computer Science and Engineering
∗†G M Institute of Technology, Davangere
Email: santoshkumarm@gmit.ac.in, preetiganiger1111@gmail.com, chavanpriyankal605@gmail.com, rakshithaboragallappanavara@gmail.com, shravyabh818@gmail.com

DOI : https://doi.org/10.55248/gengpi.07.1026.35004

ABSTRACT—

The growing sophistication of digital image-manipulation algorithms requires efficient techniques for distin-guishing real images from manipulated ones. This paper proposes a deepfake detection system that simultaneously employs three image-classification models, namely ResNet18, EfficientNet-B0, and Swin Transformer Tiny (Swin-T), to identify real and fake images. ResNet18 and EfficientNet-B0 are used to extract local features from the input image, including facial texture information, whereas Swin-T has been utilized to capture global dependencies using shifted-window attention. Each model indi-vidually predicts whether the input face image is real or fake with corresponding confidence scores, and the overall classification result is determined by an ensemble module that aggregates the model outputs. The effectiveness of the proposed approach is benchmarked on the [dataset name] and evaluated in terms of accuracy, precision, recall, F1-score, and ROC-AUC. The results of experiments will be reported after completing the training and testing procedures

I. INTRODUCTION

Digital images and videos are used extensively for convey-ing information. With developments in artificial intelligence, it is becoming easier to create or manipulate a facial image making it appear realistic. These images are known as deep-fakes. They can be created in a number of ways including face-swapping, facial expression alteration, or generating a fake face. As the quality of these images increases it can be difficult to tell if an image or video is real based off visual cues. This can cause concerns regarding the authenticity of digital media. As the image or video may not include any information about where it came from, the viewer will not know if the image portrays a real person/event. Using automated techniques can help detect if an image/video is a deepfake. This can be done by analysing visual cues which may not be visible to the human eye. However detecting deepfakes can be challenging as there may not be much visual evidence of manipulation. This may be due to the image/video being compressed, scaled, low resolution or being processed. Generalisation is also still an issue, as the model may not generalise well to new data if it has only been trained on one dataset or type of manipulation.link.springer+1

The detection of deepfakes is generally a binary classifica-tion problem. Given a face image as an input, the model must predict if it is a REAL image (not manipulated) or a FAKE image (manipulated/synthetic). The model will learn patterns from the data given in the training set. This will allow the model to classify new images. The model may also give a confidence score, this should be viewed as the output from the model.

There are many datasets to aid the development of a facial manipulation detection model. An example is Face-Forensics++, which contains manipulated videos created using various methods including DeepFakes, Face2Face, FaceSwap and NeuralTextures. It is important to have a range of manip-ulation types within the dataset to avoid the model learning features specific to certain types of creation/processing. It has been emphasised the importance of having consistent documentation regarding how the dataset was created and how it was evaluated.arxiv+

This project suggests a classification model to detect deep-fake images. Three models were used in this project including ResNet18, EfficientNet-B0 and Swin Transformer Tiny (Swin-T). The first two models are convolutional neural networks which learn visual features through convolutional layers. The models aim to extract local features from the image including textures, edges and facial features. The Swin-T model uses a transformer based approach using shifted window attention. This means it can learn the relationships between regions of the image provide different approaches

II. RELATED WORKS

In recent years, research into deepfake detection has moved away from simple classification problems and towards more robust and generalizable methods. Most research into state of the art solutions revolve around feature extraction from either the faces within videos or the video frames. Several

benchmark papers have been published highlighting the impact of data preparation and testing. Below is the section of literature review for our IEEE paper. Adapt it to reflect what we found in our project.

A. Deepfake Data Sets and Benchmarks

A popular data set for obtaining images and video clips with manipulated faces is FaceForensics++. (Ro¨ssler et al., 2019) This data set contains original and manipulated videos using four different methods to generate the deepfakes: Deep-Fakes, Face2Face, FaceSwap, NeuralTextures. The data set also contains videos with differing levels of compression to help investigate the effect it has on detection.

FaceForensics++ can help researchers determine whether their model has the capability to identify real and fake faces. However, training a detector using this data set may not produce a universal solution. If a model performs well with a specific split of the data set, it may not perform as well with another data set or with different manipulations. Therefore, when presenting results it is important to state what data set, split method and conditions were used.

B. CNN Detection

Convolutional Neural Networks (CNN) are often used for classification problems within Computer Vision. They work by using filters to detect certain features within images. These can be patterns or textures such as facial edges or cracks on skin. ResNet18 and EfficientNet-B0 are two architectures of CNNs that can be used for Deepfake Detection.

These allow you to create a classifier that can learn to separate fake faces from real faces within the data set. ResNet has residual blocks which help to detect long range patterns. EfficientNet has been designed with scalable linear units to help create a balance between increasing how deep the model is while keeping the cost low. However, it cannot guarantee that all manipulations will be picked up and are only suitable for the data we have.

C. Transformer Detection

Vision Transformers use the self attention mechanism to help classify images. Unlike CNNs they do not require local filters. Instead they group patches of images together to detect long range relationships. Swin Transformers include shifted patches at different points in the network to detect local and global patterns.

This was not considered by the authors but could help analyse the features of the CNN. We could compare the ResNet18 and EfficientNet-B0 models with the Swin-T model to see if there is consistency between the results. Comparing the Swin-T transformer with the other models would give an understanding of how complementary they are.

D. Ensemble Methods

Combining different models to create an ensemble helps to improve performance by combining the results. This can be done by taking the average of the class scores or voting. To get the best performance from an ensemble all models should perform as well as possible on the task. CNNs and Vision Transformers detect different types of artefacts so combining them will help to detect more subtle patterns within images. If the models being used are highly similar then the ensemble shouldn't outperform the individual models. For example if they are just different weights of a CNN.

Previous research into ensemble methods for deepfake de-tection states that it is important to select the correct combina-tion method and hyperparameters for the particular problem. This paper discusses how well ensembles generalise. When using an ensemble method for this task you must understand that it is just combining the results from your specific models. For example if we were to use an ensemble of ResNet18, EfficientNet-B0 and Swin-T then we could take the average of the real/fake scores from the output layer. We should compare the ensemble method to individual models using the same test set. The weights of the ensemble classifier should be tuned using the validation set.

E. Evaluation and Generalisation

It can be difficult to fairly compare different methods due to differing data pipelines, splits and evaluation metrics. DeepfakeBench aims to help this by providing an open source data handling platform, pre-existing methods and standardised evaluation methods and metrics. The DeepfakeBench frame-work contains multiple data sets and detection methods. This helps to enable reproducible research by standardising the evaluation methods.

Generalisation can also be an issue when evaluating differ-ent detection methods. A classifier may not generalise well with different data sets or compression methods. If we are testing our model with multiple data sets we should provide more information about how well it performed. This can be done by providing precision, recall, F1-score, ROC-AUC and confusion matrices. If we are only using one data set then we should state that our results only apply to this data set and aren't generalisable to other data sets or real world scenarios.

III. Input Dataset

The input data consist of real and fake images. The real images are taken from genuine photographs. While the fake images are created using deep learning techniques like Gener-ative Adversarial Networks, face swapping methods and image generating AI methods. The popular dataset used for deepfake image detection are FaceForensics++, Deepfake Detection Challenge (DFDC) Dataset, Celeb-DF and 140K Real and Fake Faces Dataset. The dataset is then preprocessed to make all images of the same size and the pixel values are normalized. A label is also added to each image with 0 being

real and 1 being fake. The images are then split into training, validation and testing datasets to create and test a deep learning model which will accurately detect if the images are real or fake.

The dataset is an important part to getting accurate results. You can collect images from public datasets. These should be separated into two folders, one for real images and one for fake images. Common image types like JPG, JPEG and PNG can be used as input. Preprocessing is done to make all images the same size, e.g. 224 x 224 pixels. They are also put into RGB format. Image augmentation can be carried out to help the model recognise more image patterns. This can be done through flipping, rotating and changing brightness. The dataset is then inputted into deep learning algorithms like Convolutional Neural Networks (CNN). This will learn features from the images and will be able to classify them as real or fake. The model can then be tested with metrics like accuracy, precision, recall and F1-score.

IV. System Overview

A. Input Face Image

- The system accepts a facial image as input.
- The image may be a real photograph or a manipu-lated/deepfake image.
- The input image is provided for authenticity analysis.
- It serves as the starting point of the detection process.

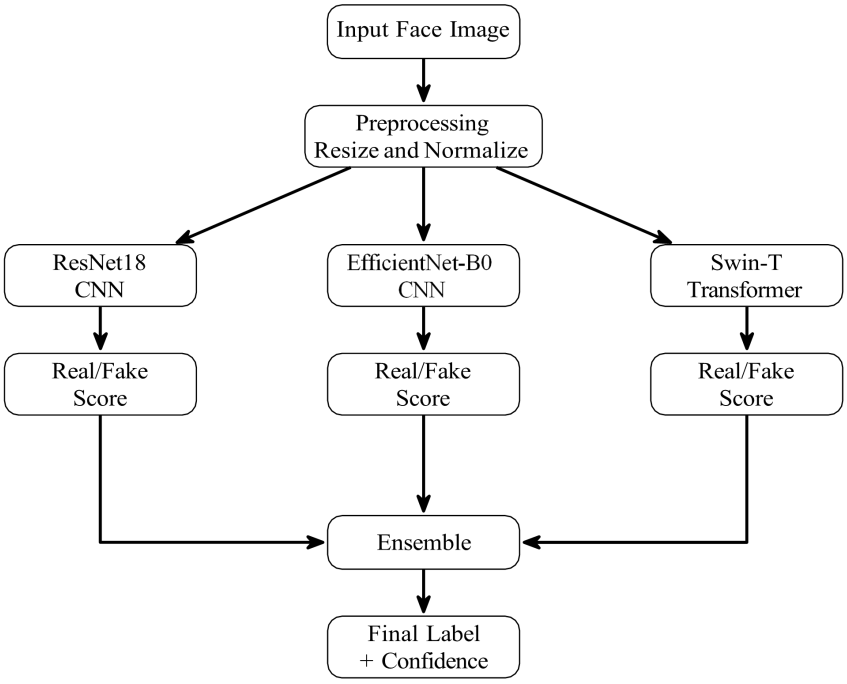


Fig. 1. Proposed system flowchart for deepfake image detection.

B. Preprocessing

- Resizes the input image to the dimensions required by the models.
- Normalizes pixel values according to the models' require-ments.
- Optionally crops the face region if face detection is implemented.
- Ensures that the image format is compatible with all three models.

C. ResNet18 (CNN)

- ResNet18 stands for Residual Network with 18 layers.
- It is a Convolutional Neural Network (CNN).
- Extracts visual features such as textures, edges, and facial patterns.
- Uses residual connections to improve information and gradient flow during training.

- Produces a prediction score for real or fake classification.

D. EfficientNet-B0

- EfficientNet-B0 is an efficient convolutional neural net-work.
- Extracts discriminative features from the facial image.
- Balances classification performance with computational efficiency.
- Uses a different architecture from ResNet18 to learn complementary features.
- Produces its own Real/Fake prediction score.

E. Swin-T (Transformer)

- Swin-T stands for Swin Transformer Tiny.
- It is a Vision Transformer architecture.
- Divides the image into visual representations and pro-cesses them hierarchically.
- Uses window-based and shifted-window self-attention to learn relationships between image regions.
- Produces its own Real/Fake prediction score

A. Real/Fake Score

- Each of the three models generates a classification output.
- The output indicates whether the image is likely to be real or fake.
- Depending on the implementation, it may be a probabil-ity, a pair of class probabilities, or raw model scores.
- These outputs are passed to the ensemble block for combining predictions.

B. Ensemble

- Combines the predictions from ResNet18, EfficientNet-B0, and Swin-T.
- Uses the aggregation method implemented in the system, such as probability averaging, weighted averaging, or majority voting.
- Integrates predictions from different model architectures
- Produces a single final prediction.
- Its effectiveness depends on the models' performance and the ensemble method.

C. Final Label + Confidence

- Displays the final classification result.
- The predicted label is either Real or Fake.
- Provides an associated confidence score or probability.
- Makes the prediction understandable to the end user.
- he confidence score should not automatically be inter-preted as the probability that the prediction is correct unless the model has been appropriately calibrated.

V. Result and Discussion

+ Table I shows two example predictions from the proposed

TABLE I

Deepfake Image Detection Results

Image Type	**Predicted Result**	**Confidence**
AI-generated image	Fake	53%
Real photograph	Real	91%

deepfake image detection system. The AI-generated image was classified as fake with a confidence score of 53%, while the real photograph was classified as real with a confidence score of 91%. These confidence scores represent the model's predictions for the two examples and should not be interpreted as the overall accuracy of the model. Overall accuracy must be evaluated using a labelled test dataset. Figure 3 presents two example outputs of the proposed detection system. The first example is classified as fake with a confidence score of 53%, while the second is classified as real with a confidence score of 91%. These examples illustrate the system's predictions on individual inputs. They do not establish overall model accuracy; this must be evaluated using a labelled test dataset.

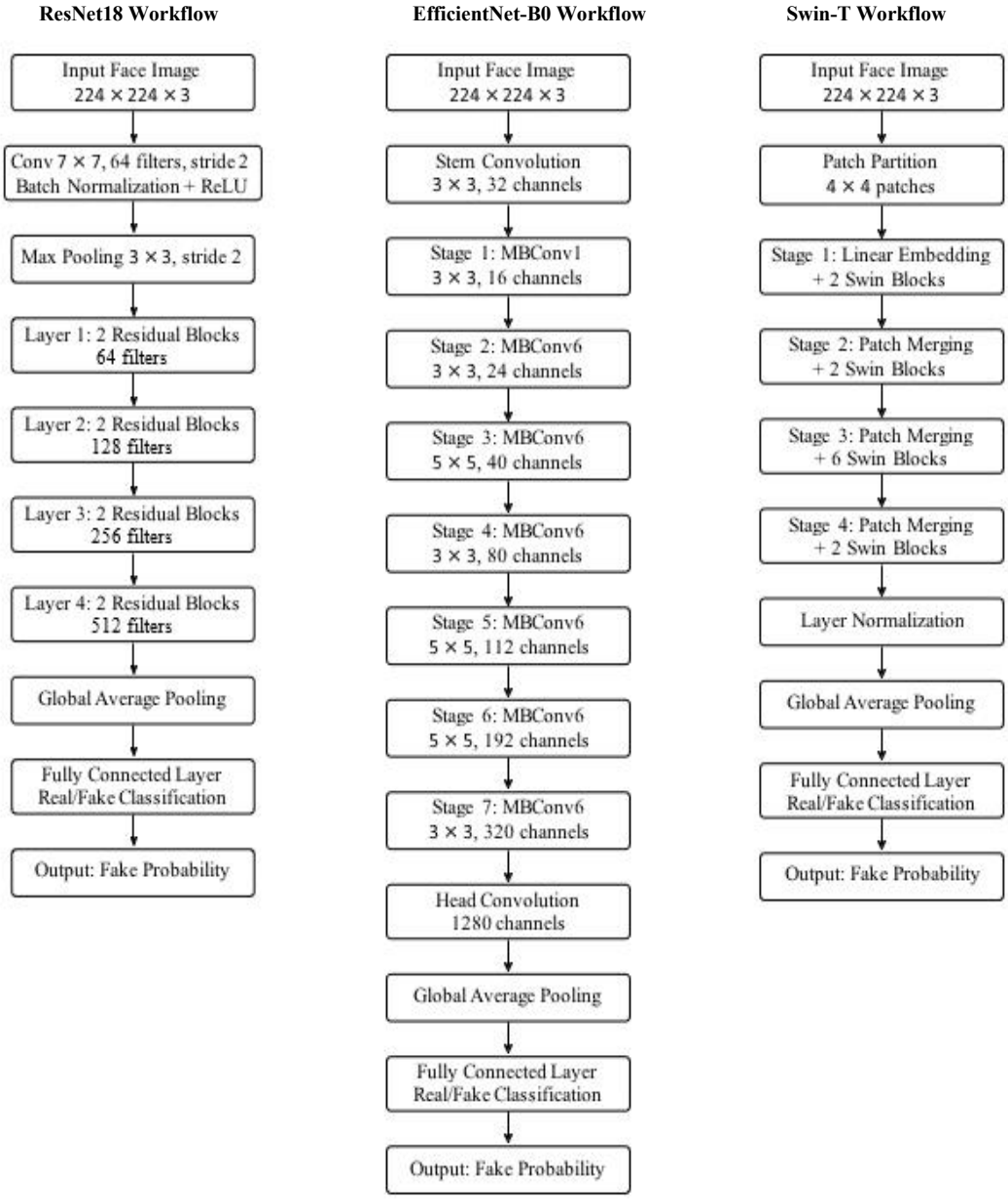


Fig. 2. Architectural workflows of ResNet18, EfficientNet-B0, and Swin Transformer-T for deepfake image detection.

Acknowledgment

The authors would like to express their sincere gratitude to their institution and faculty members for their valuable guidance, encouragement, and support throughout this re-search. Their suggestions and feedback helped improve the development and presentation of this work. The authors also acknowledge the researchers and developers who have con-tributed to deep learning architectures, including ResNet18, EfficientNet-B0, and Swin Transformer, which support the proposed deepfake detection approach. Finally, the authors appreciate the availability of relevant research publications and datasets that facilitated the study of facial image authenticity detection.

Conclusion

This paper proposes a Deep Learning based method for detecting deepfake facial images. We utilise a combination of three models (ResNet18, EfficientNet-B0, Swin Transformer (Swin-T)) to detect if an input image is a real or fake facial image. This is achieved by pre-processing the input, passing it through the three models and using an ensemble technique to combine each models prediction together.



(a) Fake image prediction: 53% confidence



(b) Real image prediction: 91% confidence

Fig. 3. Example outputs of the proposed deepfake image detection system.

Using a mixture of CNNs and Transformers can enable us to get a better idea of whether the image is real or fake. Using the Ensemble technique can provide us with a way to combine models together. By using multiple models that contribute additional information we may improve our detection. This will be shown by testing on appropriate datasets.

The next step of this project will be testing the system with various real and fake images. We will look to improve the generality of the system. We will test the performance using metrics such as Accuracy, Precision, Recall and F1-Score. We can also look to improve the efficiency and confidence score of the system which will help improve its reliability.

REFERENCES

[1] Jain, H. (2025). Digital Image Forgery Detection using Deep Learning. International Journal for Research in Applied Science and Engineering Technology.

[2] Selva Birunda, S., Nagaraj, P., Krishna Narayanan, S., Muthamil sudar, K., Muneeswaran, V., & Ramana, R. (2022). Fake Image Detection in Twitter using Flood Fill Algorithm and Deep Neural Networks. 2022 12th International Conference on Cloud Computing, Data Science & Engineering (Confluence), 285-290.

[3] Sabah, H. (2022). Detection of Deep Fake in Face Images Using Deep Learning. Wasit Journal of Computer and Mathematics Science.

[4] Rodr´ıguez-Ortega, Y., DoraM.Ballesteros, L., & Renza, D. (2021). Copy-Move Forgery Detection (CMFD) Using Deep Learning for Image and Video Forensics. Journal of Imaging, 7.

[5] A. Dosovitskiy et al., “Image classification with vision transformers,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2020.

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 779–788.

[7] Z. Zhang and L. Zhang, “Digital image forensics: A survey of methods and applications,” J. Comput. Sci. Technol., 2018.

[8] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jegou, “Training data-efficient image transformers & distillation through attention,” in Proc. Int. Conf. Mach. Learn. (ICML)., 2021, pp. 10347–10357.D.

[9] Thies, J., Zollhofer, M., Stamminger, M., Theobalt, C., & Nießner, M. (2016). Face2face: Real-time face capture and reenactment of rgb videos. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2387-2395).

[10] Korshunova, I., Shi, Ws., Dambre, J., & Theis, L. (2017). Fast faceswap using convolutional neural networks. In Proceedings of the IEEE international conference on computer vision (pp. 3677-3685).

[11] Google, ”Gemini API Documentation,” Google AI for Developers. [Online]

[12] Google, ”Google Maps Platform Documentation,”[online]

[13] T. Zhao, X. Xu, M. Xu, H. Ding, Y. Xiong, and W. Xia, “Learning selfconsistency for deepfake detection,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021.

[14] L. Chen, Y. Zhang, Y. Song, L. Liu, and J. Wang, “Self-supervised learning of adversarial example: Towards good generalizations for deepfake detection,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 18710–18719.

[15] K. Shiohara and T. Yamasaki, “Detecting deepfakes with self-blended images,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 18720–18729.

[16] A. Jain, N. Memon, and J. Togelius, “A dataless faceswap detection approach using synthetic images,” in 2022 IEEE International Joint Conference on Biometrics (IJCB). IEEE, 2022, pp. 1–7.

[17] P. Korshunov and S. Marcel, “Deepfakes: a new threat to face recogni-tion? assessment and detection,” arXiv preprint arXiv:1812.08685, 2018.

[18] A. Kohli and A. Gupta, “Detecting deepfake, faceswap and face2face fa-cial forgeries using frequency cnn,” Multimedia Tools and Applications, vol. 80, pp. 18461–18478, 2021.

[19] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Niessner, “Face2face: Real-time face capture and reenactment of rgb videos,” in Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2387–2395, 2016.

[20] M. Zendran and A. Rusiecki, “Swapping face images with generative neural networks for deepfake technology–experimental study,” Procedia computer science, vol. 192, pp. 834–843, 2021.