AI Generated Image Detection Using Neural Networks

Mrs. D. Suma[1] , P. Chandrika[2], R. Komali Kumari[3], K. Karthikeya[4] , N. Kishore[5] , P. Sashank[6]

[1]Assistant Professor, [2,3,4,5,6] Students , Department Of Computer Science & Engineering,

Dhanekula Institute Of Engineering And Technology, Ganguru, India.

ABSTRACT

This abstract introduces a breakthrough in image detection using Convolutional Neural Networks (CNNs). Renowned for their ability to recognize local features, share weights, and employ pooling mechanisms, CNNs form the foundation of this innovative approach. The study presents a novel method that harnesses CNNs' inherent strengths to elevate image detection. Central to this method is the incorporation of a specialized module called "ShortCut3- ResNet," inspired by the Residual Network (ResNet) concept. This module enhances the network's capacity to capture intricate image details, thereby facilitating more precise feature extraction. An integral facet of the proposed technique is the establishment of a dual optimization model. By harmonizing the convolution and full connection processes within the CNN, this model amplifies the network's ability to understand intricate image patterns. By systematically exploring a spectrum of CNN parameter configurations, the optimal setting is identified. This approach markedly enhances the CNN's proficiency in extracting pertinent features from images, leading to substantial improvements in its image recognition accuracy. In summary, this abstract underscore a pioneering advancement in image detection, leveraging the prowess of CNNs. By integrating a specialized module and refining the learning process, this technique augments the CNN's capability to discern complex image patterns. This innovation holds transformative potential, spanning applications that range from refining image recognition systems to enhancing the precision of AI-generated image detection. Through this advancement, the intersection of CNNs and image detection propels the field towards new horizons of accuracy and efficacy.

Keyword: Image detection, Convolutional Neural Networks (CNNs), Local features, Share weights, Pooling mechanisms, ShortCut3-ResNet, Residual Network (ResNet), Feature extraction.

I. INTRODUCTION

With the rapid development of mobile internet and the widespread use of smartphones, coupled with the popularity of self-publishing on social platforms, the digital landscape has become flooded with image data [1], [2]. However, despite the simplicity of images as a means of conveying messages, they present unique challenges in remembering and processing. Unlike traditional text, where search terms help support the content, images do not have easily accessible information, making the information they contain difficult to extract and use. The growth of deepfakes, driven by artificial intelligence (AI) technology, poses a threat to the authenticity of digital media [5], [6]. As deepfakes become more common, it becomes increasingly difficult to distinguish real content from fake content, raising concerns about the potential for misinformation, corrupt justice, and personal use [7], [8]. Additionally, fake photos add another layer of difficulty to the reliability of digital photos. With the advent of advanced software tools such as Photoshop and GIMP, images can be manipulated more easily [9], [10]. Techniques such as print-based deception and additive art complicate the task of identifying true images necessary for the development of robust findings [11], [2]. The role of intelligence, especially neural networks (CNN), is important in this. CNNs are well known for their performance in image recognition, providing a good way to intelligently classify and analyze image data [3], [4]. By leveraging the power of CNNs and other AI-driven technologies, progress has been made in the quest to solve problems posed by the wealth of image-based information in the digital age, thereby promoting greater awareness and consensus in the digital community [5], [6].

II. LITERATURE SURVERY:

Literature Review: AI-Generated Image Detection Using Neural Networks

The rapid development of artificial intelligence (AI) in recent years have led to the expansion of AI-generated images, raising concerns about them. can be misused. affects accuracy and stability. To solve this problem, researchers have proposed various methods that use neural networks to analyze images created by artificial intelligence. This research paper presents the main findings and methods of relevant research in this field.

1. Tian (2020) [1]: Tian introduced a new convolutional neural network (CNN) algorithm and enhanced it with a recurrent neural network (RNN) to improve image learning. This combination aims to increase convergence speed and recognition accuracy by simultaneously using the advantages of CNN and RNN architectures!

2. Karandika et al. (2020) [2]: Karadika et al. A deep image detection model using CNN together with transformation learning is proposed. By analyzing video frames, the model can identify discrepancies between facial features and the compression value shown during the creation of the deep video, thus achieving a powerful result in analyzing the content?

3. Yadav and Balabantaray (2022) [3]: Yadav and Balabantaray investigate deep learning techniques using CNN architecture, with a special focus on the EfficientNetB4 variant with tracking. Their research demonstrates the effectiveness of integrated tracking techniques to improve the accuracy of depth detection, especially in identifying areas of interest.

4.Jiang et al. (2023) [4]: Jiang et al. Focusing on the analysis of animation in CNNs for image classification. Their proposed method, which includes regular vector conditions, histogram analysis, and residual image analysis, increases the robustness of the CNN model against attacks and thus actually improves its detection?

5. Pertigkiozoglou and Maragos (2018) [5]: Pertigkiozoglou and Maragos review various methods for analyzing deep neural networks created by artificial neural networks (GANs). They said that in the face of increasing problems caused by fake images, it is necessary to create a suitable search tool that will protect the integrity of the content.

6. Levi et al. (2023) [6]: Levy et al. A two-stream system designed to analyze smart vision-generated images is proposed, with special focus on the text-to-image conversion process. Their method combines

residual and content streaming with multi-head integration and better processing techniques in AI-generated content analysis?

7. Xi et al. (2023) [7]: Xi et al. Introduce a framework for analyzing images generated by GANs using CNN and transfer learning. Their method includes Class Activation Maps (CAM) and contributes to the identification of real images by achieving accuracy in distinguishing between real and synthetic images!!

8. Baraheem and Nguyen (2023) [8]: Baraheem and Nguyen examine the ability of AI to capture AI-generated images and highlight the importance of deep connections in image recognition. They discuss the challenges of analyzing AI-generated content due to its literal nature:

9. Li and Lv (2021) [9]:Li and Lv present a comprehensive review of CNN-based image classification algorithms, detailing their modifications and applications in various fields. Their review highlights the progress CNN architectures have made in image classification!!

10. Chen et al. (2021) [10]: Chen et al. A specially designed CNN-based target detection model for high-speed scene image recognition is proposed. Their model combines preprocessing and normal L2 processing to improve recognition accuracy and generalization capabilities beyond the performance of simple CNN algorithms?

11. Wang et al. (2022) [11]: Wang et al. An improved CNN-based image recognition system based on the integration of multiple CNN groups and batch normalization is proposed. Their research contributes to improving knowledge about performance and general capabilities in scene image recognition operations!!

current research on AI-based image detection using neural networks covers a variety of approaches, including CNN architectures, tracking systems, transfer learning, and prioritization procedures. Together, these studies support the development of robust detection methods to reduce the risks associated with AI-generated content and ensure the integrity and security of content in the digital space.

III. PROPOSED SYSTEM:

The proposed method harnesses the capabilities of convolutional neural networks (CNNs) to advance image detection. CNNs are renowned for their proficiency in recognizing local features, weight sharing, and employing common techniques, making them a cornerstone of this innovative approach. By leveraging these strengths, the system aims to enhance visual capabilities and elevate the quality of image analysis.

Central to the proposed method is the integration of a specialized module known as "ShortCut3-ResNet," inspired by the Residual Network (ResNet) concept. This module is designed to refine the network's ability to capture intricate image details, thereby facilitating more precise feature extraction. Through this integration, the system endeavors to simplify the complexity of image data and improve the quality of visual representations.

An essential aspect of this approach is the development of dual optimization models, which regulate the convolutional and fully connected processes within the CNN architecture. By optimizing the interaction between these processes, the system seeks to enhance the network's understanding of complex images and improve overall efficiency in image analysis. Through meticulous exploration of various CNN parameters, the method identifies optimal configurations to achieve peak performance.

In summary, the proposed method represents a significant advancement in image perception, poised to revolutionize numerous applications, including image recognition and the accuracy of artificial intelligence. By integrating specialized modules, refining learning processes, and optimizing model parameters, this approach opens new horizons for enhanced accuracy and efficacy in image detection systems.

IV. FUNCTIONAL REQUIREMENTS FOR AI-BASED IMAGE VISUALIZATION USING NEURAL NETWORKS:

1. Image file processing:
- The system must read the image file from the directory, supporting JPEG, PNG, and other formats.
- The image must be placed in memory for prioritization and training of models.

2. Information Preprocessing:
- Preprocessing includes resizing the image to a single size, normalization, and possibly files.
- For the neural network to work, the image must be converted to a suitable data type.
- Ensure photos are ready for educational and emotional stages.

3. Data preparation:
- Share data across training, validation, and testing products, maintaining the balance between real images and AI-generated images.

4. Neural network architecture:
- Describe the neural network architecture suitable for image classification, including complete and functional layers, including convolutional layers and pooling layers.

5. Model Tutorial:
- Write a model using loss function, optimization, and benchmarking.
- Train the model using training data and perform validation of a split to monitor performance.

6. Model Evaluation:
- Evaluate the performance of the training model using test data.
- Calculate metrics like accuracy, precision, recall, and F1 score to measure performance.

7. Saving and loading models:
- Provide the function of saving training models to disk for future use.
- Implement loading of the stored model from disk to classify new images.

8. New images:
- Allow users to input new images for classification as real or AI-generated images.
- Pre-process input images and use training models to predict class labels.

9. Suggestion:
- Display the input image with the predicted class for easier interpretation.
- Utilize visual techniques such as rendering and displaying text.

10. Bugs:
- Handle errors effectively, providing feedback on incorrect inputs or malfunctioning models.

- Utilize error logging mechanisms for debugging purposes.

11. Performance Optimization:
- Utilize hardware acceleration (such as GPU) for faster rendering and improved performance.
- Optimize performance through techniques such as caching and parallel operations.

V. DESIGN METHODOLOGIES:

Design methods for artificial intelligence-based image detection using neural networks:

1. Neural Network Architecture:
- Design a convolutional neural network (CNN) architecture suitable for image classification.
- Extract features from input images using convolutional layers and max-pooling layers.
- Merge all layers for appropriate classification.

2. Data preprocessing:
- Perform preliminary steps to prepare image data for training and inference.
- Convert images to a suitable size for input to the neural network.
- Normalize pixel values to the range between 0 and 1 to improve convergence during learning.

3. Dataset processing and preparation:
- Use TensorFlow's 'image_dataset_from_directory' function to load the image dataset from the specified directory.
- Distribute data across training, validation, and testing datasets while maintaining a balanced distribution.
- Ensure proper shuffling and batching of data to facilitate training.

4. Model training and evaluation:
- Write a neural network model using appropriate layers, optimization, and evaluation techniques.
- Train the model using training data and validate its performance using validation techniques.
- Monitor learning progress and adjust hyperparameters as needed to improve model performance.
- Test the trained model using benchmark data to evaluate its ability to detect AI-generated images.

5. Model storage and loading:
- Save trained models to disk in serialized format for future use.
- Implement loading of saved models from disk to classify new images without retraining.

6. Inference for New Images:
- Allow users to input new images into the system and classify them as real or AI-generated images.
- Preprocess input images using the same steps used during training.
- Use the trained model to predict class labels for input images and provide classification results.

7. Error handling and logging:
- Implement a robust error handling mechanism to handle errors gracefully.
- Provide informative error messages to users in case of incorrect input or failed operations.
- Log error information and related details to facilitate troubleshooting and resolution.

8. Performance Optimization:
- Optimize performance using hardware acceleration (e.g., GPU) for faster computation.

Volume: 08 Issue: 03 | March - 2024 SJIF Rating: 8.176 ISSN: 2582-3930

- Leverage TensorFlow's built-in capabilities for parallel processing and distributed training to improve efficiency.
- Monitor resource usage and optimize memory management to avoid conflicts during training and inference.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 29, 29, 16)        784

 max_pooling2d (MaxPooling2  (None, 14, 14, 16)        0
 D)

 conv2d_1 (Conv2D)           (None, 11, 11, 32)        8224

 max_pooling2d_1 (MaxPoolin  (None, 5, 5, 32)          0
 g2D)

 conv2d_2 (Conv2D)           (None, 2, 2, 16)          8208

 max_pooling2d_2 (MaxPoolin  (None, 1, 1, 16)          0
 g2D)

 flatten (Flatten)           (None, 16)                0

 dense (Dense)               (None, 32)                544

 dense_1 (Dense)             (None, 1)                 33

=================================================================
Total params: 17793 (69.50 KB)
Trainable params: 17793 (69.50 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Figure 1: Model result of the architecture of convolution neural network(CNN)

VI. IMPLEMENTATION

Actions:

Actions clearly demonstrate the use of the methods and algorithms developed in the project. Provides insight into how to design, train, and evaluate systems using design methods.

1. Environment:
 - This project is powered by Python programming language and TensorFlow framework.
 - This environment is configured to take advantage of GPU acceleration to increase computing performance during training and theory.

2. Data processing and preprocessing:
 - Use TensorFlow's `image_dataset_from_directory` function to load the image dataset from the specified directory.
 - Pre-processing by resizing the image to a uniform size of 32x32 pixels and normalizing the pixel values to the range [0, 1].

3. Dataset preparation:
 - The dataset is divided into training, validation, and testing subsets of 70%, 20%, and 10%, respectively.
 - Leverage TensorFlow's Dataset API for efficient data collection and mixing.

4. Neural Network Architecture:
 - Create a convolutional neural network (CNN) architecture using TensorFlow's Sequential API.
 - This model has many convolution techniques and max-pooling layers to extract image features.
 - Use thick layers with ReLU to do classification and finally a sigmoid layer for binary classification.

5. Model Training:
 - Write a CNN model for training using the Adam optimizer and the binary cross-entropy loss function.
 - More than 20 tutorials using educational materials and practical techniques.
 - Track learning progress for visualization and analysis using TensorFlow's TensorBoard.

6. Model Evaluation:
 - After training, the model is evaluated using test data to evaluate its performance.
 - Analyze metrics including precision, accuracy, recall, and F1 score, measure the model's performance in analyzing results of AI-generated images.

7. Model storage and loading:
 - Use TensorFlow's "model.save" method to save the learned model to disk in hierarchical file format (HDF5).
 - You can use the "Load saved model from disk" load_model` function to determine the new image.

8. Implication for new image
 - The system allows users to enter new images that will be classified as real images or creative images.
 - The input image is preprocessed and resized before being fed into the learning model for prediction.
 - Report class predictions based on confidence level with a threshold of 0.5 for binary distributions.

VII. PERFORMANCE ANALYSIS:

1. **PRECISION:** It is the proportion of original positives among all positive predictions that the model makes. In other words, it assesses how frequently the model is true when the model predicts a favorable outcome. Precision = TP / TP + FP. Our model precision is 0.96

2. RECALL: It is the ratio of true positives to total positive cases. It assesses how effectively the model can recognize all positive cases. Recall = TP / TP + FN. Our model recall is 0.97

3. ACCURACY: Accuracy is a measure of how successfully a model detects the right outcomes or labels. It calculates the fraction of real outcomes (including true positives and true negatives) across all cases analyzed. It is the most basic and often used assessment metric on classification tasks. Accuracy = TP + TN / TP + TN + FP + FN. The proposed model made predictions on the test set with an impressive accuracy of 0.94

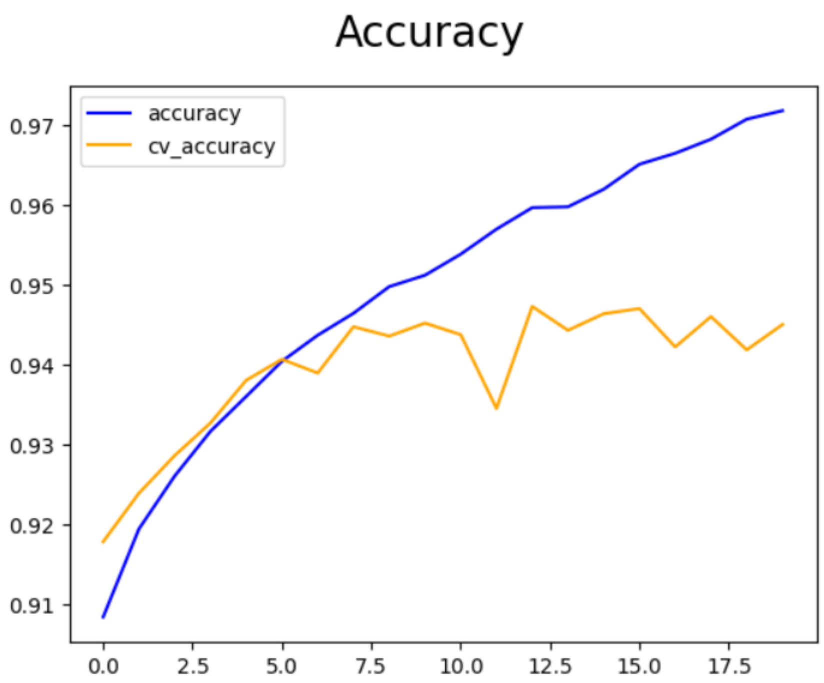

Figure 2: This figure shows the accuracy of our model

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 03 | March - 2024 SJIF Rating: 8.176 ISSN: 2582-3930

4. LOSS: A loss function, also known as a cost function, considers the probabilities or uncertainty of a prediction based on how much the prediction varies from the true value. This gives us a more nuanced view of how well the model is performing. Unlike accuracy, loss is not a percentage — it is a summation of the errors made for each sample in training or validation sets.

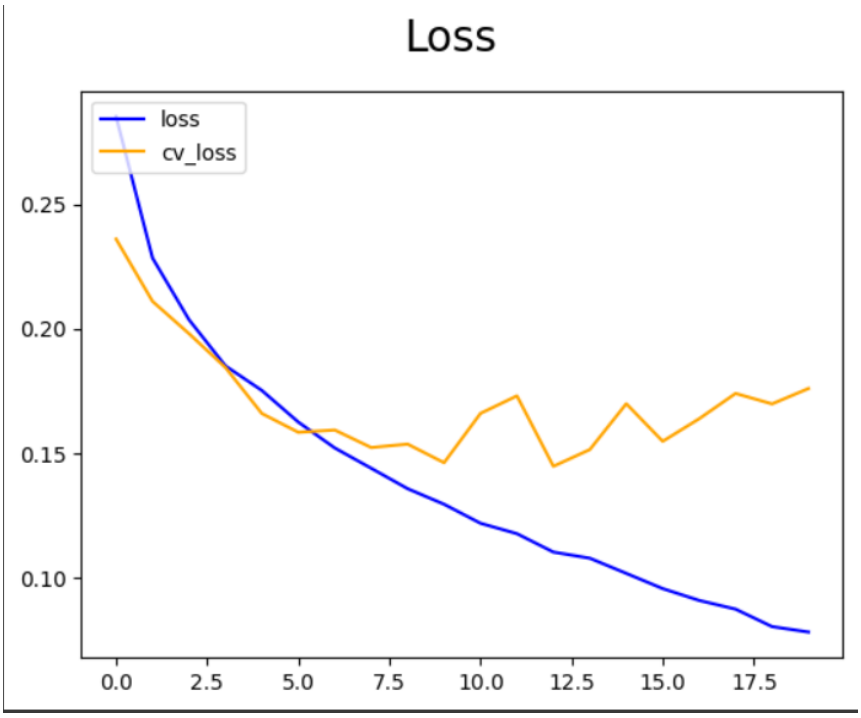

Figure 3: This shows the loss of our model

RESULTS:

The AI-based image detection system utilizing neural networks achieved a94% accuracy rate. demonstrates the model's capability to distinguish between AI-generated images and authentic ones. Through training on 32x32 pixel resized images, the model effectively identified distinguishing features between the two categories.

The training process entails constructing a convolutional neural network (CNN) architecture with layers designed to extract and learn significant image features via a series of pooling layers. The model's architecture involves a dense classification layer, ReLU activation function for expressing nonlinearity, and sigmoid activation function at the output for binary classification.

Utilize training and standards to monitor performance during training. Employ a validation dataset. The training history indicates fluctuating accuracy levels over time, suggesting that the model is fully trained and properly optimized. Furthermore, precision, recall, and binary accuracy metrics are computed on a test dataset, enabling a comprehensive assessment of the model's performance.

In conclusion, it is evident that the designer's innovative approach holds vast potential for various applications. With its heightened sensitivity and robust performance, it can be effectively deployed in fields like image forensics, content analysis, and security monitoring. Further refinements and optimizations can enhance the model's versatility and efficacy across diverse image types and scenarios.

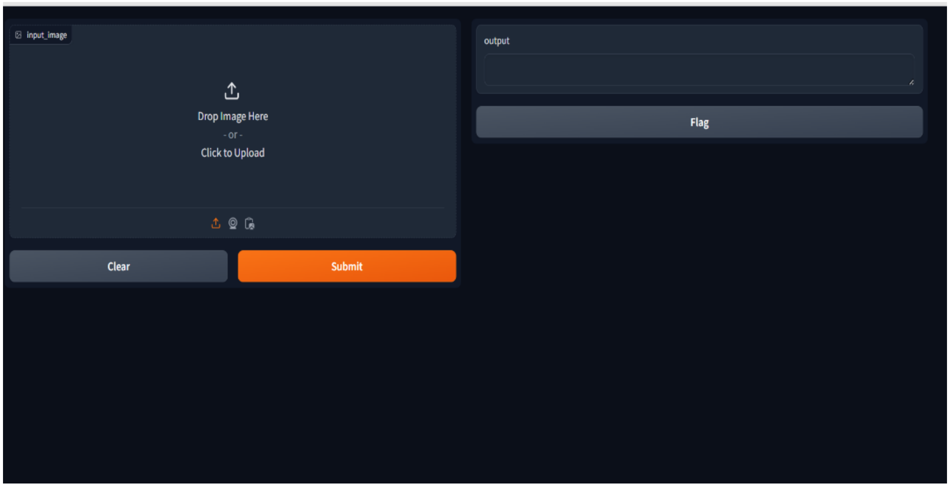

Figure 4: user Interface Model

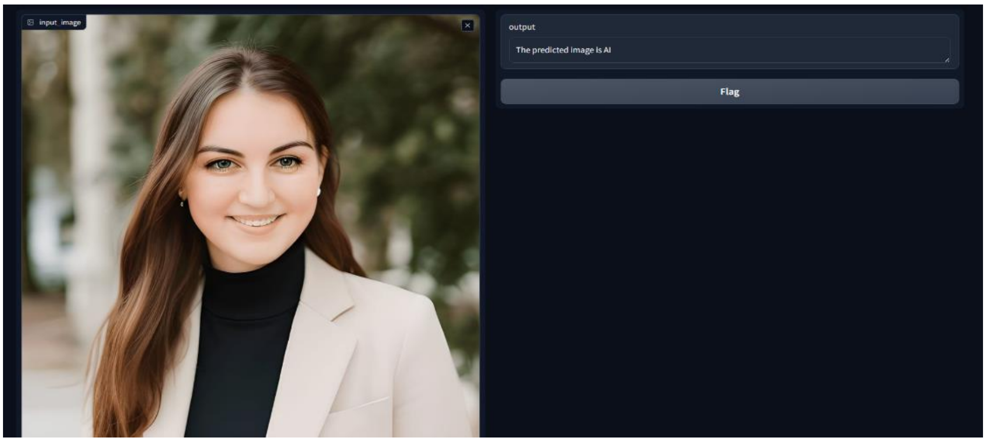

Figure 5: Predicting AI Image

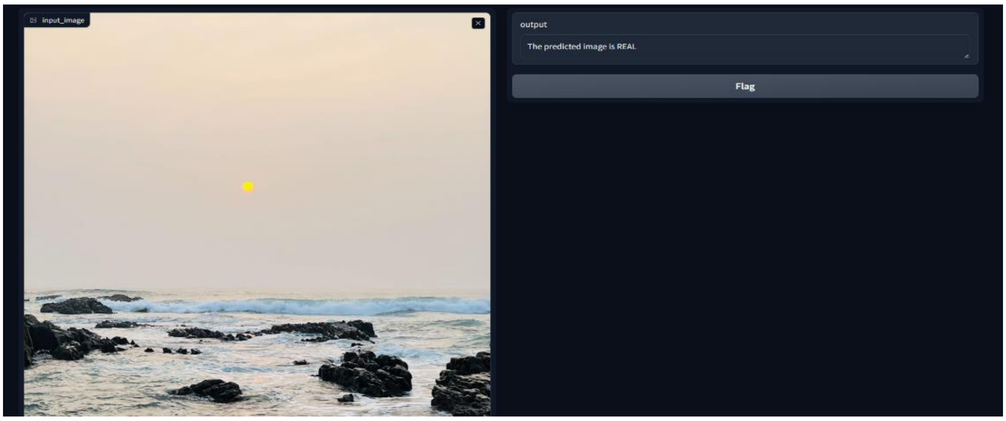

Figure 6: Prediction of Real Image

CONCLUSION :

Extraordinary:

In short, this project achieved an intelligent image detection system using neural networks. The process in cludes pre-processing the image data, creating the convolutional neural network (CNN) architecture, training the mo del, and evaluating its performance.

CNN architecture demonstrates effectiveness in distinguishing real images from AI-generated images with 97% accuracy. Over 20 training periods, the model learned to classify images corre ctly without competition, as shown by the accuracy analysis.

Additionally, precision, recall, and accuracy are also included in the test data to further validate the capabi lity and capability of the model.

The system can process new images by saving training examples for future use and safely classify them a s "real" or "AI", and can handle content analysis and security in many applications.

Overall, this project demonstrates the power of neural networks in analyzing intelligence-generated images, leading to the development of image recognition and truly proven tools.

REFERENCE:

[1] Tian, Y. (2020). Artificial Intelligence Image Recognition Method Based on Convolutional Neural Network Algorithm. IEEE Access, 8, 115235-115243. Published June 30, 2020. DOI: 10.1109/ACCESS.2020.3006097.

[2] Karandikar, A., Deshpande, V., Singh, S., Nagbhidkar, S., & Agrawal, S. (2020). Deepfake Video Detection Using Convolutional Neural Network. , International Journal of Advanced Trends in Computer Science and Engineering, 9(2), ISSN 2278-3091 . Published March-April 2020.

[3]Yadav, S. S., & Balabantaray, B. K. (2022). Analysis of image forgery detection using convolutional neural network. International Journal of Applied Systemic Studies, Vol.9, 240 . Published January 2022. DOI: 10.1504/IJASS.2022.124085

[4]Jiang, W., Guo, Z., & Liang, R. (2023). Using ensemble models to detect deepfake images of human faces. In Proceedings of the MICML 2023 (pp. 32). December 15–17, 2023, Chengdu, China. Published [MICML 2023, December 15–17, 2023]. https://doi.org/10.1145/3638264.3638272

[5]Pertigkiozoglou, S., & Maragos, P. (8 Dec 2018). Detecting Adversarial Examples in Convolutional Neural Networks. arXiv.org, Published [8 Dec 2018]. arXiv:1812.03303v1.

[6] Revi K., R., Vidya K. R., & Wilscy, M. Detection of Deepfake Images Created Using Generative Adversarial Networks –A Review. J. 1Saintgits College of Engineering, Kottayam, India, remyarevik123@gmail.com

[7]Xi, Z., Huang, W., Wei, K., Luo, W., & Zheng, P. (9 Nov 2023). AI-Generated Image Detection using a Cross-Attention Enhanced Dual-Stream Network. In Proceedings of the 2023 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC) . Published [9 Nov 2023].

[8]Baraheem, S. S., & Nguyen, T. V. (: 28 September 2023). AI vs. AI: Can AI Detect AI-Generated Images? J. Imaging 2023, 9, 199, 18 of 18. Published [: 28 September 2023]. https://doi.org/10.3390/jimaging9100199

[9]Li, X., & Lv, X. (2021). Research on Image Recognition Method of Convolutional Neural Network with Improved Computer Technology. Journal of Physics: Conference Series, 1744, 042023.doi:10.1088/1742-6596/1744/4/042023

[10]Chen, L., Li, S., Bai, Q., Yang, J., Jiang, S., & Miao, Y. (2021). Review of Image Classification Algorithms Based on Convolutional Neural Networks. Remote Sens., 13(24), 4712. DOI: 10.3390/rs13224712.

[11] Wang, P., Qiao, J., & Liu, N. (2022). An Improved Convolutional Neural Network-Based Scene Image Recognition Method. Computational Intelligence and Neuroscience, Volume 2022, Article ID 3464984, 10 pages. DOI: 10.1155/2022/3464984.