INNOVATION LAB PROJECT

MADE BY-

SOMIL AGGARWAL(2201AI36) AND SWATHI KEERTHANA(2201AI38)

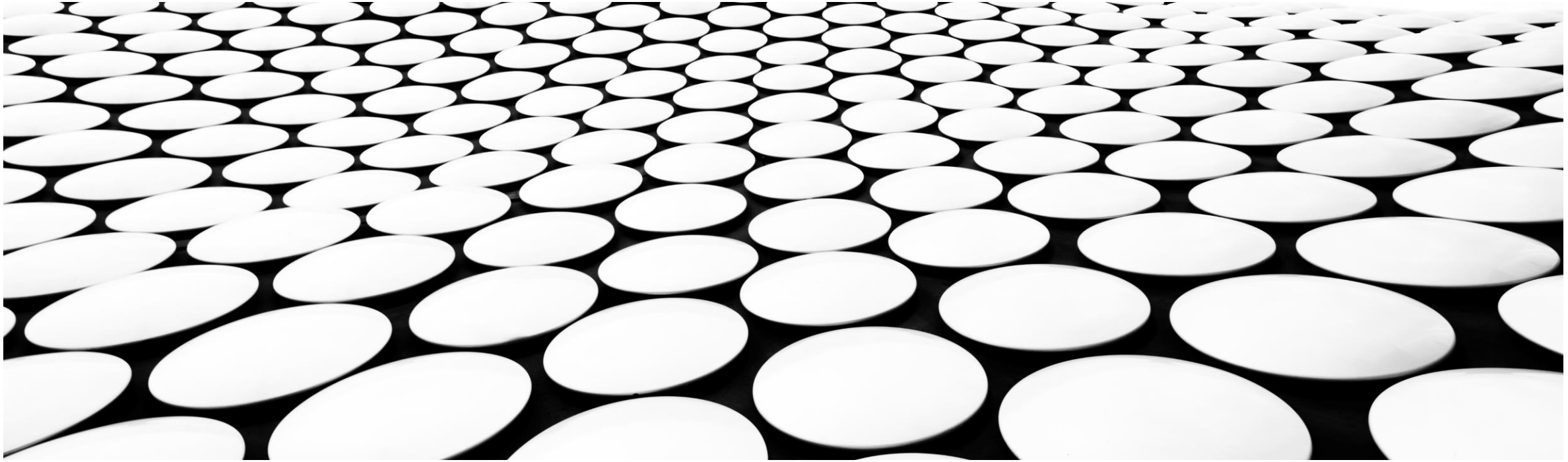
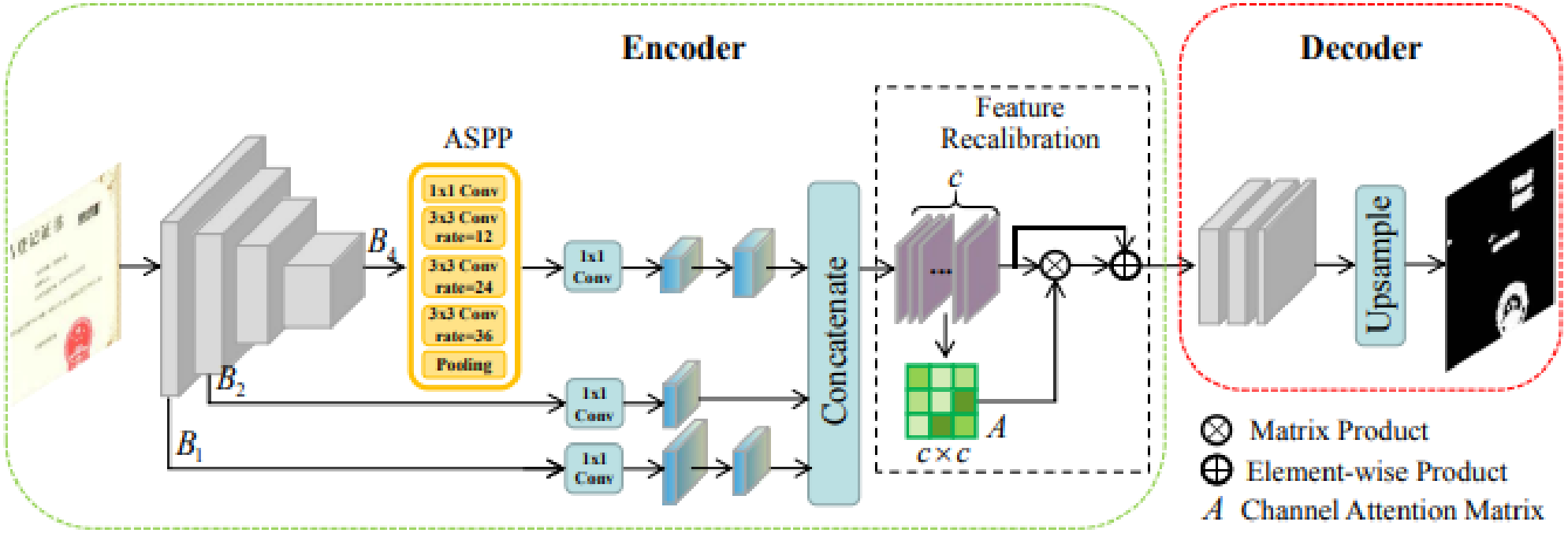


IMAGE FORGERY DETECTION MODEL

The pervasive influence of digital media has ushered in an era where the authenticity of visual information is increasingly challenged. Malicious actors exploit image manipulation techniques to spread misinformation, fabricate fake news, and perpetrate illegal activities. To counteract this growing threat, we have developed a cutting-edge model capable of detecting and pinpointing forged regions within digital images.

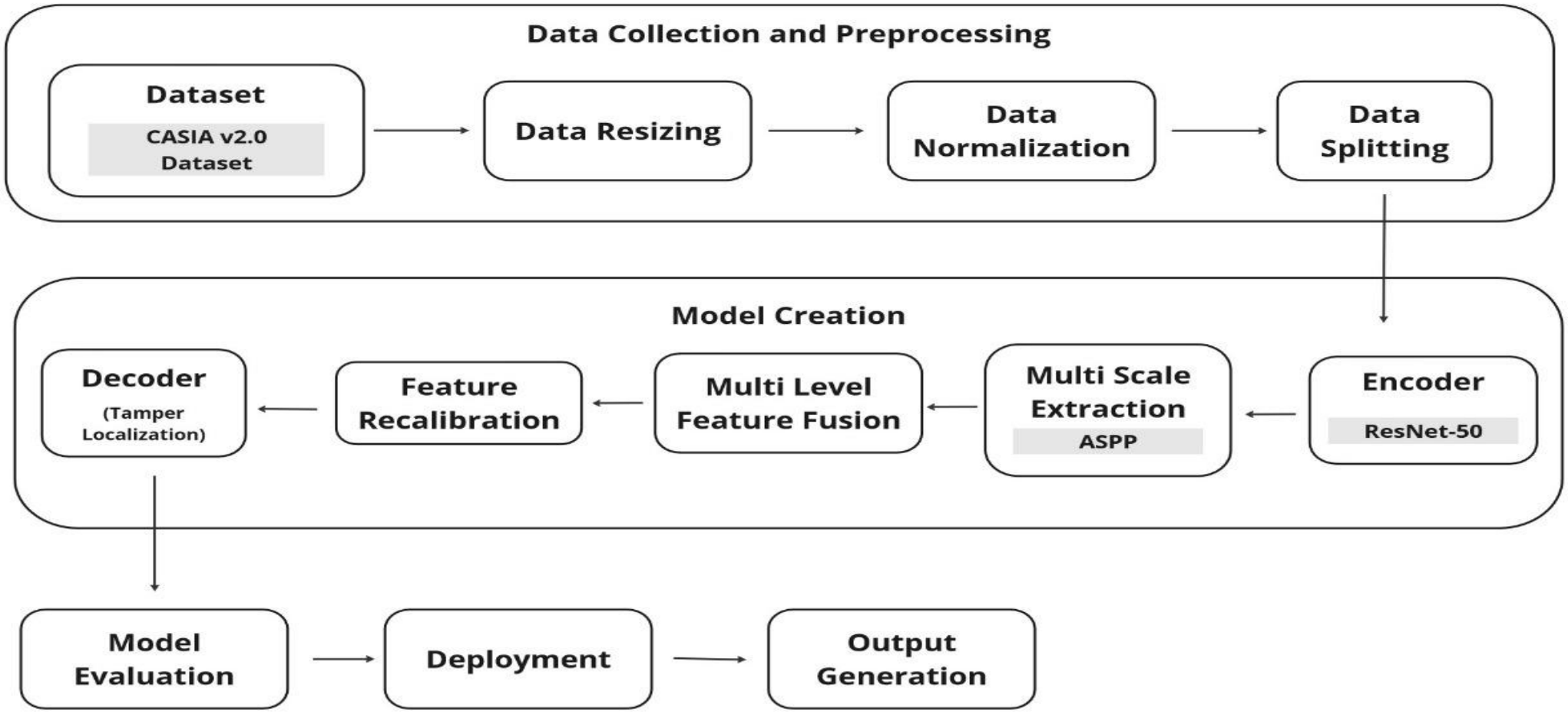
Our model employs sophisticated image processing and machine learning and deep learning algorithms to scrutinize images for signs of tampering. By identifying and highlighting the specific areas where alterations have been made, we empower individuals and organizations to critically assess the veracity of visual content. This innovative approach serves as a powerful tool in the fight against digital deception, safeguarding the integrity of information and promoting responsible use of digital media.

ARCHITECTURE OF OUR MODEL



Multi Feature Attention Network(MFAN)

WORKFLOW


Data Collection and Preprocessing
Dataset
CASIA v2.0 Dataset
Data Resizing
Data Normalization
Data Splitting
Model Creation
Encoder
ResNet-50
Multi Scale Extraction
ASPP
Multi Level Feature Fusion
Feature Recalibration
Decoder
(Tamper Localization)
Model Evaluation
Deployment
Output Generation

WHY MULTI FEATURE ATTENTION NETWORK(MFAN)??

The proposed MFAN model addresses the challenges of detecting fake certificate images with varying tampered region sizes and multiple manipulation types. Key features include:-

- **Handles Variable Tampered Regions:** The model effectively addresses the varying sizes of tampered regions, from single letters to larger areas.
- **Detects Multiple Manipulation Types:** It can identify various types of manipulations within a single image.
- **Leverages Multi-Scale Features:** Atrous Convolution captures multi-scale information, improving the detection of both large and small tampered regions.
- **Preserves Local Information:** Incorporating low-level features helps retain crucial local details, especially for smaller objects.
- **Feature Recalibration:** A feature recalibration module enhances the representation of tampered regions, enabling the model to focus on relevant information.
- **Encoder-Decoder Architecture:** The MFAN architecture effectively extracts features and generates accurate localization masks.

PREPROCESSING AND DATA VISUALIZATION FOR IMAGE FORENSICS

•**Preprocessing:**

- Images are resized to a consistent size (e.g., 512x512 pixels).
- Ground truth masks (indicating tampered regions) are converted to binary format.
- Error handling ensures processing continues despite potential issues like missing files.

•**Data Visualization:**

- A separate function helps locate the corresponding ground truth mask for a tampered image.
- Another function displays a sample tampered image alongside its mask, providing a visual reference for the dataset.

These steps prepare the data for our model and enable you to visually inspect tampered images and their corresponding forgery locations.

DATASET ORGANIZATION AND SPLITTING

Dataset Organization:

• **Mapping:** To ensure accurate pairing of tampered images with their corresponding ground truth masks, a mapping process is employed.

•**DataFrame:** The mapped data is structured into a Pandas DataFrame for efficient data management and analysis.

•**Columns:** The DataFrame includes columns for the filenames and paths of both tampered images and their corresponding ground truth masks.

Data Splitting:

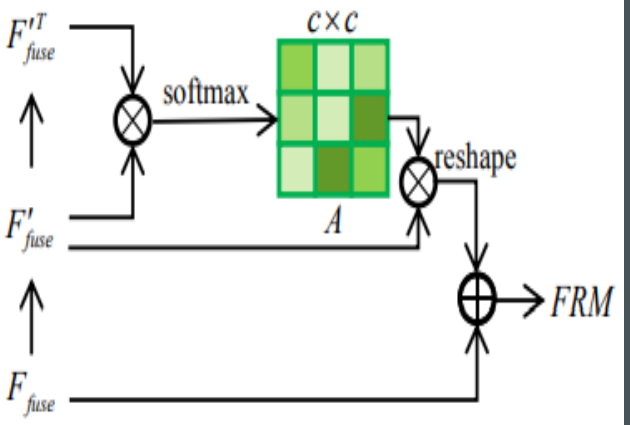
•**Stratified Split:** The dataset is split into training, testing, and validation sets using stratified sampling to maintain class distribution.

•**Randomization:** Randomization ensures that the splits are unbiased and representative of the overall dataset.

•**Proportions:** The data is typically split into a 70-30 ratio for training and testing/validation, and then the 30% is further divided equally for testing and validation.

This organized approach facilitates efficient training and evaluation of the image forensics model.

ENCODER NETWORK FOR IMAGE FORGERY DETECTION



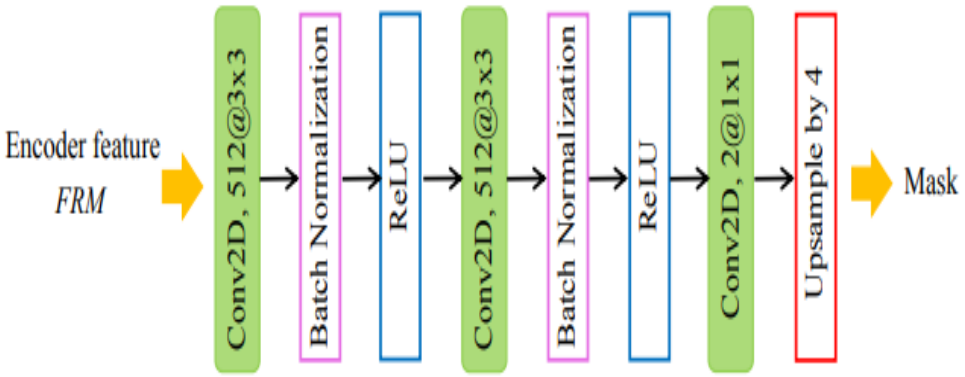
(a) Structure of feature recalibration module.

- The encoder network is a crucial component of the image forgery detection model. It extracts meaningful features from the input image, which are then used to identify tampered regions. This network typically employs a pre-trained convolutional neural network (CNN) like ResNet-50 as its backbone.
- Employs Atrous Spatial Pyramid Pooling (ASPP) to capture multi-scale contextual information, improving the model's ability to detect manipulations at various scales.

- Key features of the encoder include:
- **Multi-Scale Feature Extraction:** By utilizing atrous convolution and combining features from different layers, the encoder captures information at various scales. This helps in detecting both large and small-scale manipulations.
- **Feature Recalibration:** A feature recalibration module is used to enhance the representation of tampered regions. This involves adjusting the weights of different feature channels to emphasize relevant information and suppress noise.

The encoder network provides a robust foundation for the subsequent detection process, enabling accurate localization of tampered areas within the image.

DECODER NETWORK: RECONSTRUCTING FORGERY MAPS



The Structure of Decoder Network

- The decoder network is responsible for generating the final forgery map, which highlights the tampered regions in the image. It takes the high-level features extracted by the encoder and gradually upsamples them to the original image size.
- Key aspects of the decoder include:
- **Feature Upsampling:** The decoder progressively upsamples the feature maps to match the input image size.
- **Convolutional Layers:** Convolutional layers are used to refine the features and predict the forgery map pixel-wise.
- **Activation Functions:-** ReLU activation is applied to introduce non-linearity and improve the model's ability to learn complex patterns.
- The final output of the decoder is a pixel-wise binary mask where "1" indicates a tampered region and "0" indicates an authentic region.

LOSS FUNCTIONS USED IN OUR MODEL

The loss function L_{total} in MFAN is composed of localization loss and auxiliary loss. The localization loss L_{loc} is the binary cross-entropy loss calculated between the prediction mask and the ground truth label:

$$L_{loc} = y_{gt} log(y_p) + (1 - y_{gt}) log(1 - log(y_p))$$

where $y_{gt} = 1$ if the pixel is tampered, otherwise $y_{gt} = 0$, and y_p is the prediction mask. The auxiliary loss L_{aux} is added after the third block of ResNet-50, which helps to optimize the learning process. It is also a binary cross-entropy loss defined as follows:

$$L_{aux} = y_{gt} log(y_b) + (1 - y_{gt}) log(1 - log(y_b))$$

where y_b is the result of feature map B3 after several convolution layers and resizing. L_{loc} takes the major responsibility, while Laux is used to assist the network with training. In order to balance the importance between them, the weight α is added to Laux:

$$L_{total} = L_{loc} + \alpha L_{aux}$$

TRAINING AND EVALUATION OF OUR MODEL

Training:

•The MFAN model is trained using the Adam optimizer and a custom loss function that combines binary cross-entropy loss with an auxiliary loss to encourage accurate localization.
•A learning rate scheduler is employed to adjust the learning rate during training, optimizing convergence.

Evaluation:

•The trained model is evaluated on a held-out test set to assess its performance.
•Key metrics, including accuracy, precision, recall, and F1-score, are used to evaluate the model's ability to detect tampered regions accurately.

CHALLENGES AND SOLUTIONS IN IMAGE FORGERY DETECTION

Challenge: Computational Resource Constraints

•**Large Dataset:** The CASIA2 dataset is extensive, requiring significant computational resources for training.
•**Training Time:** Training deep learning models on large datasets can be time-consuming, especially without dedicated hardware.

Solution: Leveraging GPU Acceleration

•**GPU Power:** GPUs are highly parallel processors optimized for numerical computations, making them ideal for training deep neural networks.
•**Accelerated Training:** By transferring the model and dataset to a GPU-enabled environment, we can significantly reduce training time.
•**Enhanced Performance:** GPUs allow for faster convergence and improved model performance.

CONTINUED...

Current Progress:

•**Model Accuracy:** Our current model achieves a promising accuracy of 90% on the CASIA2 dataset.

•**Training Epochs:** The model is trained for 150 epochs to ensure robust learning and generalization.

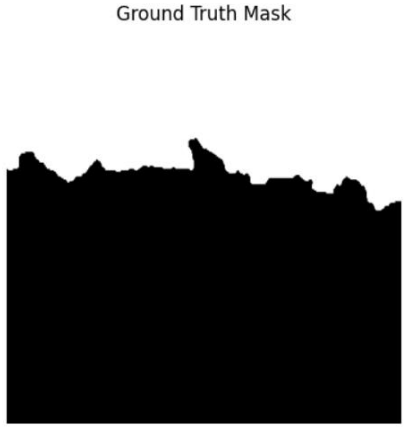
By addressing the computational challenges and leveraging GPU acceleration, we are confident in further improving the accuracy and efficiency of our image forgery detection model.

SOURCE CODE OF OUR PROJECT–
https://github.com/mswathi04/Image-Forgery-Detection

SOME OUTPUT SNIPPETS OF OUR MODEL

```
Epoch [1/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [2/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [3/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [4/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [5/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [6/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [7/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [8/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [9/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [10/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [11/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [12/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [13/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [14/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [15/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [16/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [17/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [18/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [19/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [20/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [21/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [22/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [23/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [24/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [25/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [26/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [27/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [28/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [29/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [30/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [31/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [32/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [33/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [34/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [35/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [36/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [37/50], Loss: 11.1269, Accuracy: 90.73%
Epoch [38/50], Loss: 11.1269, Accuracy: 90.73%
```

```
Accuracy for sample 0: 0.9942
Accuracy for sample 1: 0.7391
Accuracy for sample 2: 0.9824
Accuracy for sample 3: 0.9859
Accuracy for sample 4: 0.9938
Accuracy for sample 5: 0.6501
Accuracy for sample 6: 0.9347
Accuracy for sample 7: 0.9661
Accuracy for sample 8: 0.9885
Accuracy for sample 9: 0.9798
Accuracy for sample 10: 0.5274
Accuracy for sample 11: 0.9237
Accuracy for sample 12: 0.9419
Accuracy for sample 13: 0.9976
Accuracy for sample 14: 0.8863
Accuracy for sample 15: 0.9791
Accuracy for sample 16: 0.9042
Accuracy for sample 17: 0.9731
Accuracy for sample 18: 0.9933
Accuracy for sample 19: 0.9862
Accuracy for sample 20: 0.5177
Accuracy for sample 21: 0.8083
Accuracy for sample 22: 0.6192
Accuracy for sample 23: 0.9910
Accuracy for sample 24: 0.9941
Accuracy for sample 25: 0.9902
Accuracy for sample 26: 0.9288
Accuracy for sample 27: 0.8551
Accuracy for sample 28: 0.9905
Accuracy for sample 29: 0.4055
Accuracy for sample 30: 0.9634
Accuracy for sample 31: 0.8931
Accuracy for sample 32: 0.9984
Accuracy for sample 33: 0.9198
Accuracy for sample 34: 0.5003
Accuracy for sample 35: 0.8888
Accuracy for sample 36: 0.8073
```



FUTURE SCOPE OF OUR MODEL

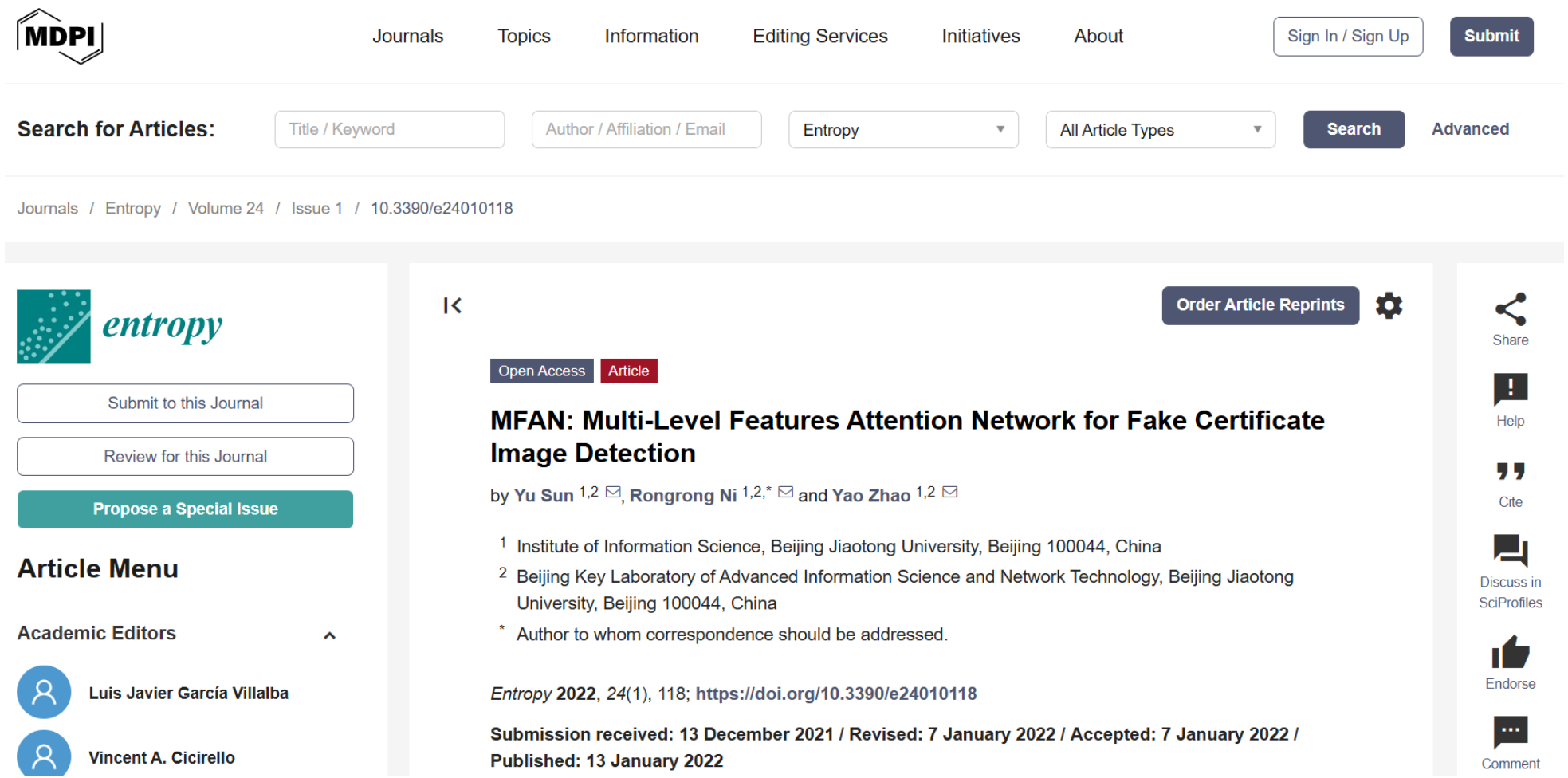
•**Real-Time Forgery Detection:** We aim to optimize our model for real-time applications, enabling the detection of forged images as they are captured or shared.

•**Adversarial Attacks and Defenses:** We will investigate the vulnerability of our model to adversarial attacks and develop techniques to make it more robust.

•**Forgery Localization and Quantification:** We plan to refine our model to precisely localize the tampered regions and estimate the extent of manipulation.

•**Type of forgery:** We can also determine the type of forgery that has been done to the tampered image. E.g.- Splicing, Color change, Copy-move etc.

REFERENCES


MDPI
Journals
Topics
Information
Editing Services
Initiatives
About
Sign In / Sign Up
Submit
Search for Articles:
Title / Keyword
Author / Affiliation / Email
Entropy
All Article Types
Search
Advanced
Journals / Entropy / Volume 24 / Issue 1 / 10.3390/e24010118
entropy
Submit to this Journal
Review for this Journal
Propose a Special Issue
Article Menu
Academic Editors
Luis Javier García Villalba
Vincent A. Cicirello
Order Article Reprints
Open Access
Article
MFAN: Multi-Level Features Attention Network for Fake Certificate Image Detection
by Yu Sun 1,2, Rongrong Ni 1,2,* and Yao Zhao 1,2
1 Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China
2 Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing Jiaotong University, Beijing 100044, China
* Author to whom correspondence should be addressed.
Entropy 2022, 24(1), 118; https://doi.org/10.3390/e24010118
Submission received: 13 December 2021 / Revised: 7 January 2022 / Accepted: 7 January 2022 / Published: 13 January 2022
Share
Help
Cite
Discuss in SciProfiles
Endorse
Comment