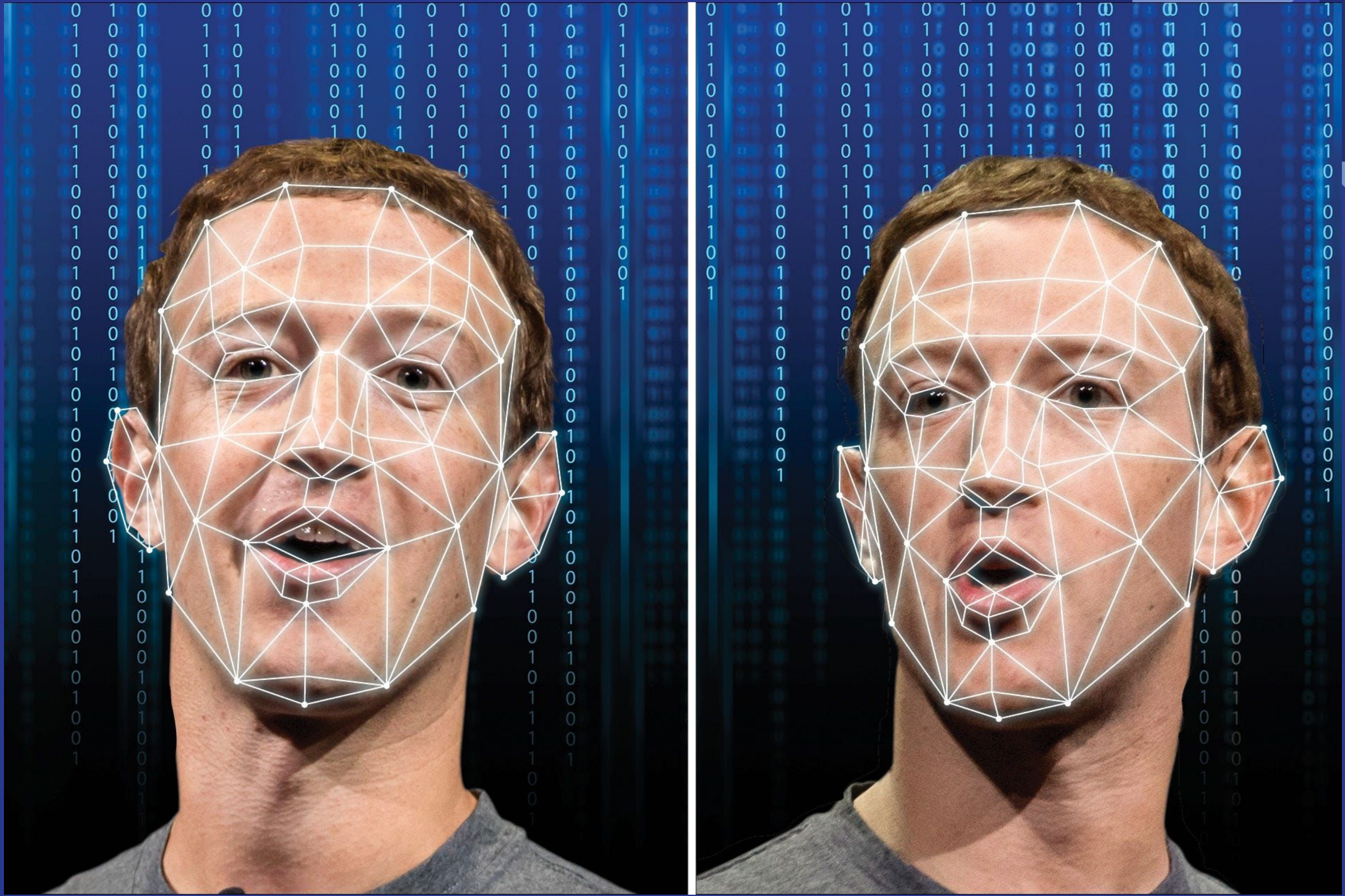
# Deepfake Detection Using Deep Learning Techniques

Authors: Sayantan Bhattacharyya, Milind Chakraborty, Nitin Sharma

**Guide: Prof. Dharmendra Singh Rajput**



# Introduction

**Research Topic:** Deepfake Detection Using Deep Learning Techniques
**Problem:**

- Deepfake videos threaten individual and national security.
- They manipulate public opinion, spread misinformation, and endanger individuals.
- Current detection methods are limited in accuracy and effectiveness.

**Objective:**

- Investigate and develop robust techniques for detecting deepfake videos.
- Utilize advancements in vision transformers and inception net technology for accurate detection.

**Importance:**

- Critical need for innovative solutions in combating deepfake threats.
- Developing a reliable detection method is crucial for safeguarding individuals and strengthening national security.

# Literature Review

**Overview:**

- Summarizes key findings from relevant research papers.

**Methods and Results:**

- Various approaches for deepfake detection explored by researchers.
- CNN, LSTM, VGG network, optical flow, and dense units utilized for frame feature extraction, image augmentation, and residual conversion.

**Accuracy and Performance:**

- Different models achieved varying levels of accuracy.
- Ranging from 75.46% to 97.1% depending on the methodology and dataset used.

**Significance:**

- Literature highlights the ongoing efforts to develop effective deepfake detection methods.
- Provides valuable insights for informing our own research approach and methodology.

**References:**

- Citations of relevant research papers for further reading and validation of findings.

# Literature Review - Citation 1

**Authors: D. Güera and E. J. Delp**
**Methodology:**

- Used CNN and LSTM for frame feature extraction and temporal sequence analysis.
- Shallow network with two fully-connected layers and one dropout layer.

**Dataset:**

- Contains 600 deepfake videos from multiple sources and the HOHA dataset.

**Accuracy:**

- Achieved 97.1% accuracy with 80 frames.

**Significance:**

- Demonstrates effectiveness of CNN and LSTM in deepfake detection.
- Provides a strong baseline for comparison with other methods.

# Literature Review - Citation 2

**Authors: X. Chang et al.**
**Methodology:**
- Proposed a VGG network based on noise and image augmentation.
- Utilized an SRM filter layer and image augmentation layer.

**Dataset:**
- Trained and evaluated on the Celeb-DF dataset.

**Accuracy:**
- Achieved an accuracy of 85.7%.

**Significance:**
- Introduces innovative approach using noise and augmentation for detection.
- Shows promising results on a widely used dataset.

# Literature Review - Citation 3

**Authors: Huaxiao Mo et al.**
**Methodology:**
- Converted RGB images into residuals and passed through convolutional layers.
- Used three-layer groups with convolutional layers, LReLu activation, and max pooling.

**Dataset:**
- Prepared from the CELEBA HQ dataset.

**Accuracy:**
- Actual accuracy not mentioned in provided information.

**Significance:**
- Highlights a unique approach of converting images into residuals for detection.
- Provides insights into leveraging architectural designs for deepfake detection.

# Literature Review - Citation 4

**Authors: Irene Amerini, Leonardo Galteri, Roberto Caldelli, Alberto Del Bimbo**

**Methodology:**

- Used optical flow and CNN pre-trained with VGG-16/ResNet50.
- Utilized sigmoid activation to determine frame authenticity.

**Dataset:**

- Utilized the FaceForensics++ dataset.

**Accuracy:**

- Achieved 81.61% accuracy with VGG16 and 75.46% with ResNet50.

**Significance:**

- Demonstrates the use of optical flow for deepfake detection.
- Provides insights into the effectiveness of different CNN architectures.

# Literature Review - Citation 5

**Authors: Hsu, Chih-Chung, Yi-Xiu Zhuang, Chia-Yen Lee**
**Methodology:**
- Proposed a CFFN consisting of dense units with transition layers and a growth rate.
- Utilized a convolution layer with 128 channels and 3x3 kernel size.

**Dataset:**
- Utilized a dataset extracted from CelebA.

**Accuracy:**
- Achieved a recall value of 0.900.

**Significance:**
- Introduces a novel architecture for deepfake detection.
- Shows promising recall values for identifying manipulated images.

# Literature Review - Citation 6

**Authors: Hasin Shahed Shad et al.**

**Methodology:**

- Employed basic CNN architecture and pre-trained models using DenseNet and ResNet iterations.
- Dataset consisted of 70,000 genuine faces and one million fake faces.

**Accuracy:**

- Achieved an accuracy of 81.6% with ResNet50.

**Significance:**

- Demonstrates the effectiveness of pre-trained models for deepfake detection.
- Provides insights into handling large-scale datasets for training.

# Literature Review - Citation 7

**Authors: Theerthagiri P, Basha Nagaladinne**
**Methodology:**

- Utilized the InceptionNet Convolutional Neural Network (CNN) algorithm for deepfake detection.
- Different types of transitions in real images were used for testing.

**Dataset:**

- Utilized the DFDC dataset.

**Accuracy:**

- Achieved an overall accuracy of 93%.

**Significance:**

- Highlights the effectiveness of InceptionNet for deepfake detection.
- Provides insights into performance metrics on real-world datasets.

# Framework

1. Image:
   Input images are fed into the model for processing.
2. Data Preprocessing:
   a. Augmentation techniques applied to enrich and diversify the training dataset:
   b. Aim: Enhance dataset diversity, improve model generalization, and enable robust deepfake detection.
3. Model Architecture:
   The model architecture extends InceptionV3 with a **flattening layer** followed by **dense layers**. It consists of **two dense layers** with 512 units and ReLU activation, each followed by a **dropout layer** (rate: 0.5). Subsequently, a **dense layer** with 64 units and ReLU activation is added, leading to a **final dense layer** with 1 unit and sigmoid activation for binary classification. The model is compiled with Adam optimizer (learning rate: 0.0001) and binary cross-entropy loss.

4. Training:
   Model is trained on the augmented dataset with Adam optimizer (learning rate: 0.0001) and binary cross-entropy loss function.
5. Hyperparameter Tuning/Testing:
   Iterative process of adjusting hyperparameters and evaluating model performance on validation and test datasets.
6. Deploy Model:
   Once trained and evaluated, the model is ready for deployment.
7. [Fake, Real]:
   Model output: Probability scores indicating the likelihood of an image being categorized as fake or real.

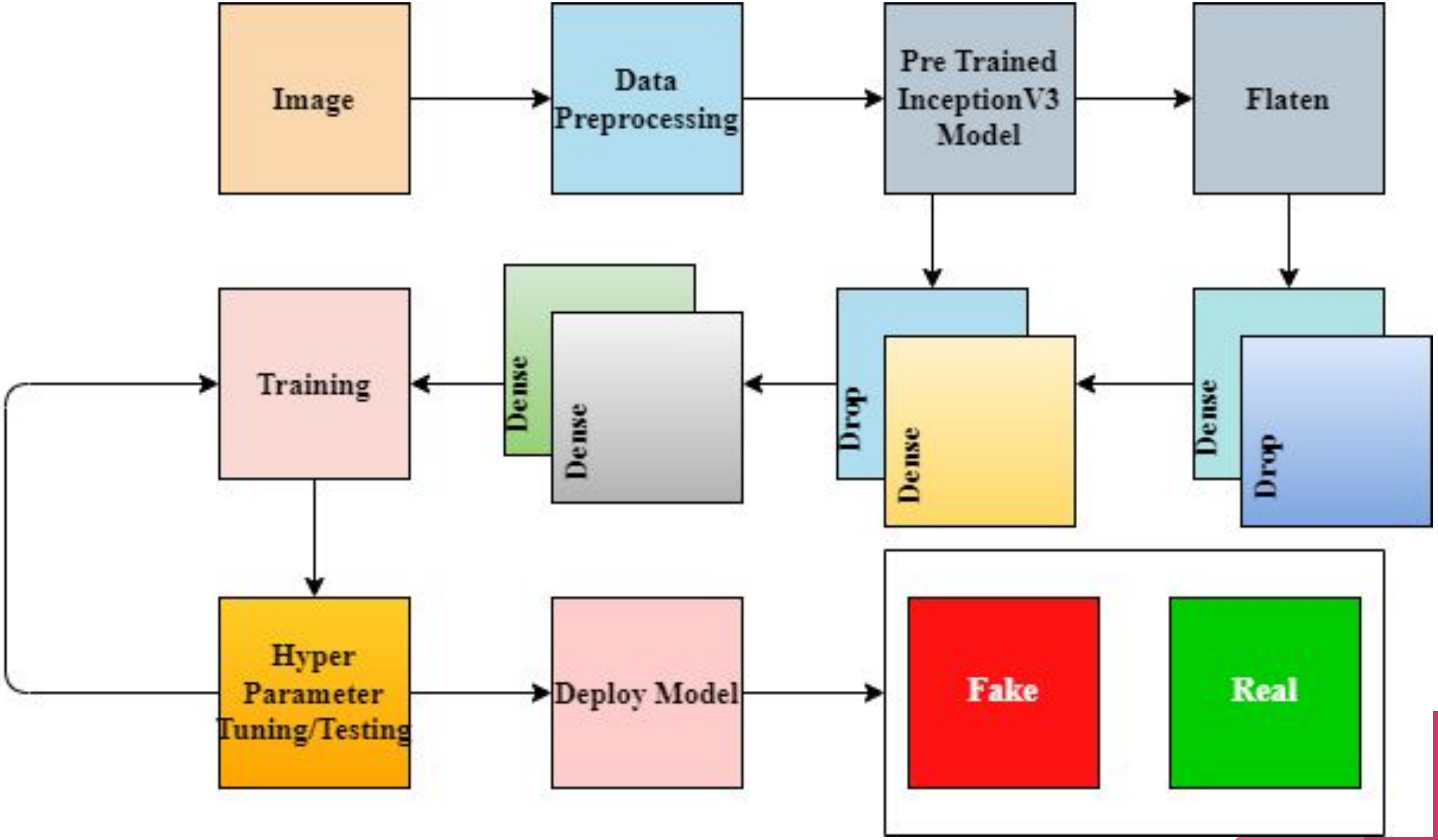


Image
Data Preprocessing
Pre Trained InceptionV3 Model
Flaten
Training
Dense
Dense
Drop
Dense
Dense
Drop
Hyper Parameter Tuning/Testing
Deploy Model
Fake
Real

# Dataset

- Meticulously Curated Dataset:
  a. Total images: 190,341
  b. Source: Kaggle
- Balanced Distribution:
  a. Real images: 70,000
  b. Fake images: 70,000
- Data Split:
  a. Training: 40,000 images
  b. Validation: 20,000 images
  c. Testing: 2,000 images
- Randomized Sampling Strategy:
  a. Ensured diverse representation.
- Prioritized Diversity:
  a. Balanced representation for nuanced understanding.
- Facilitated Precise Classification:
  a. Robustness ensured through meticulous curation.

# Preprocessing

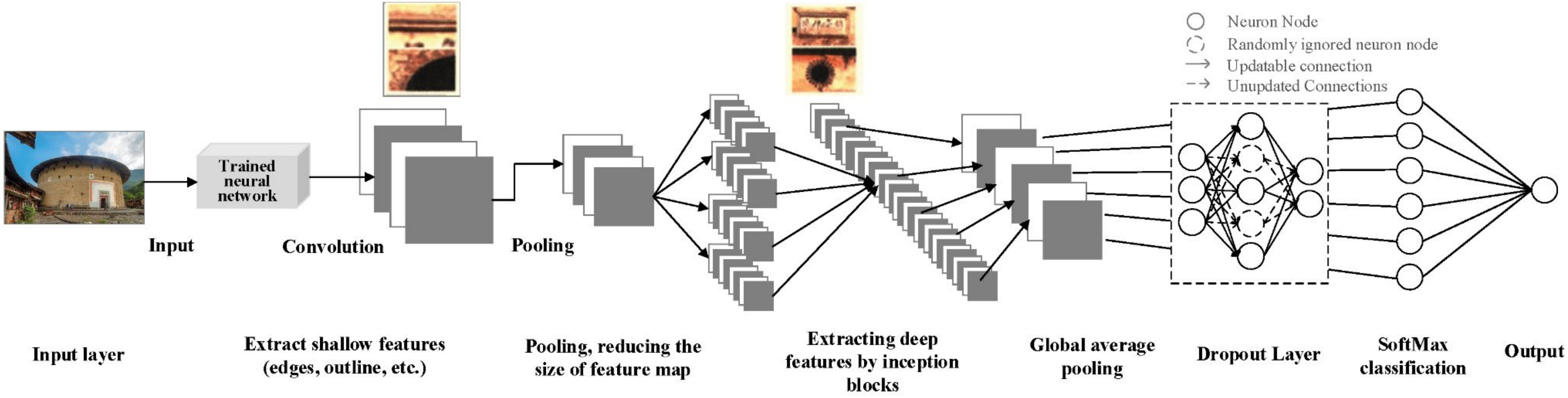
- Augmentation techniques applied to enrich and diversify the training dataset:
  - Normalization: Pixel values are normalized to a range of 0 to 1.
  - Rotation: Images are rotated within -10 to +10 degrees.
  - Shifts: Up to 10% of image width and height.
  - Shearing: Up to 20% of image width.
  - Random zooming: Within a 10% range.
  - Horizontal flipping: 50% probability.
  - Fill mode: "Nearest" used for handling new pixel introductions.
- Aim: Enhance dataset diversity, improve model generalization, and enable robust deepfake detection.

# Model Architecture

| Layer (type) | Output Shape | Param # |
|---|---|---|
| inception_v3 (Functional) | (None, 2, 2, 2048) | 21802784 |
| flatten (Flatten) | (None, 8192) | 0 |
| dense (Dense) | (None, 512) | 4194816 |
| dropout (Dropout) | (None, 512) | 0 |
| dense_1 (Dense) | (None, 512) | 262656 |
| dropout_1 (Dropout) | (None, 512) | 0 |
| dense_2 (Dense) | (None, 64) | 32832 |
| dense_3 (Dense) | (None, 1) | 65 |

Total params: 26293153 (100.30 MB)
Trainable params: 26258721 (100.17 MB)
Non-trainable params: 34432 (134.50 KB)

# InceptionV3 Model



# Training

- Defining Callbacks:
  - EarlyStopping and ModelCheckpoint callbacks are defined.
  - EarlyStopping monitors validation loss and halts training if it doesn't improve for a certain number of epochs (patience=3).
  - ModelCheckpoint saves the best model based on validation loss.
- Model Training:
  - The model is trained for 10 epochs using the `fit` method.
  - Training data is fed into the model using `train_generator`, and validation data using `val_generator`.
- Training Progress:
  - Training progress is shown with epoch-wise results.
  - Each epoch displays training accuracy, training loss, validation accuracy, and validation loss.
  - Example output illustrates the progression of accuracy and loss metrics throughout the training process.

# Evaluation Metrics

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| real | 0.95 | 0.89 | 0.92 | 1000 |
| fake | 0.89 | 0.95 | 0.92 | 1000 |
| accuracy | | | 0.92 | 2000 |
| macro avg | 0.92 | 0.92 | 0.92 | 2000 |
| weighted avg | 0.92 | 0.92 | 0.92 | 2000 |

# Comparative Analysis

1. Comparison with Existing Approaches:
   a. Our model exhibits competitive performance compared to existing deepfake detection methods.
   b. Despite some approaches achieving higher accuracies, our model demonstrates robustness and reliability, especially in scenarios with imbalanced class distributions.
   c. The simplicity and interpretability of our model make it suitable for practical deployment in real-world applications, contributing to the advancement of deepfake detection technology.
2. Discussion on Model Architecture:
   a. The proposed architecture is built upon the InceptionV3 base model, utilizing its advanced feature extraction capabilities.
   b. Dropout layers are incorporated into the model to mitigate overfitting, thereby enhancing generalization performance.
   c. The sequential arrangement of dense layers enables the extraction of hierarchical features, leading to accurate classification outcomes.

# Results

- Model Performance:
  - The proposed architecture demonstrated promising performance in distinguishing between authentic and deepfake images.
  - Overall Accuracy: 92% on the test set.
  - Precision, Recall, and F1-score metrics indicate balanced performance across both classes.
  - High Precision and Recall: The model effectively discerns manipulated content, showing high precision and recall for both real and fake images.

# Conclusion

- This study introduces a novel CNN architecture for deepfake detection, leveraging advancements in convolutional neural networks and transfer learning.
- Through meticulous dataset curation and augmentation, along with a comprehensive model architecture, we have developed a reliable solution for identifying manipulated imagery.
- The model's performance underscores its potential utility in safeguarding individuals and mitigating the adverse impacts of deepfake proliferation.

# Limitations

- Applicability solely to image data. While deepfake detection often extends to video content, our model's scope is confined to static images.
- Addressing this limitation would require the development of temporal analysis techniques tailored to video-based deepfake detection.
- Additionally, ongoing advancements in deepfake generation techniques may challenge the model's efficacy over time, highlighting the need for continuous research and adaptation in this rapidly evolving field.