Advance Attendance Management System by Implementing Several Deep Learning Models

॥वसुधैव कुटुम्बकम्॥

THESIS SUBMITTED TO

Symbiosis Institute of Geoinformatics

FOR PARTIAL FULFILLMENT OF THE M. Sc. DEGREE

By

Kinjal Bandopadhyay, Ayan Datta, Ritwik Dubey, Ayush Mitra

(PRN – 22070243028, 22070243006, 22070243038, 22070243007)

Symbiosis Institute of Geoinformatics

Symbiosis International (Deemed University) 5th Floor, Atur Centre, Gokhale Cross Road, Model Colony, Pune – 411016

ACKNOWLEDGEMENT

We would like to extend our sincere appreciation to Symbiosis Institute of Geoinformatics, Symbiosis International (Deemed University), for providing us with the opportunity and resources to complete our project titled "Advance Attendance Management System by Implementing Several Deep Learning Models."

Our deepest gratitude goes to our project guide and mentors for their invaluable guidance and support. We also thank our dedicated team members, Kinjal Bandopadhyay, Ayan Datta, Ritwik Dubey, and Ayush Mitra, for their hard work and cooperation throughout the project.

We are grateful to our friends and family for their unwavering support and understanding during this journey. Lastly, we acknowledge the academic community and researchers whose work inspired and informed our project.

Thank you to everyone who contributed to our success.

Sincerely,

Kinjal Bandopadhyay, Ayan Datta, Ritwik Dubey, Ayush Mitra

INDEX

Sl. No.	Contents	Page No.
1	Preface	4
2	Abstract	5
3	Introduction	6
4	Objective	8
5	Literature Review	9
6	Data-Set Description	14
7	Methodology	16
8	Data Collection	16
9	Model Building (From Scratch)	18
10	Model Training (From Scratch)	21
11	Utilizing Transfer Learning For Attendance Management System	23
12	Overview Of Advance Attendance Management System	34
13	Results And Discussion	38
14	Conclusion	40

PREFACE

In today's rapidly evolving world, the need for efficient attendance management is paramount for educational and corporate institutions. Traditional methods have proven inefficient and error-prone. Our project, "Building Attendance Management Models: Deep Learning and More," sets out to construct attendance models from the ground up, offering a fresh perspective on this age-old challenge.

This preface introduces the project's significance and objectives. We aim to showcase the process of building attendance management models using various deep learning techniques, including sequential modeling and transfer learning.

Our journey begins with an analysis of current attendance management systems, emphasizing their limitations. We will delve into the theory behind deep learning and its application to attendance management.

As we progress, we will provide insights into the model-building process, covering data preprocessing, architecture, training, and fine-tuning. Ethical considerations and data security protocols will also be addressed.

Our project's culmination will present empirical results, highlighting the advantages of our model-driven attendance management system over traditional methods. We invite you to join us on this journey, where we redefine attendance management with innovative model-building techniques.

Welcome to the future of attendance management, where deep learning leads the way.

ABSTRACT

In an era defined by digital transformation, it is vital for institutions and businesses to adapt and innovate to remain competitive and efficient. Among the challenges faced by academic institutions and organizations alike is the arduous task of attendance management. The traditional manual methods for recording attendance are riddled with inefficiencies and inaccuracies. They require significant administrative effort and resources, and they remain susceptible to human errors. In recognition of these limitations, this project introduces a state-of-the-art solution that harnesses the power of computer vision, deep learning, and face recognition.

The abstract is the cornerstone of a project that strives to revolutionize attendance management. Through the application of advanced technologies, including convolutional neural networks (CNNs) and the utilization of pre-trained models like InceptionResNetV2, we present a highly accurate and automated attendance tracking system. This system has the capacity to recognize individuals' faces and cross-reference them with a pre-defined database, culminating in precise attendance records.

Our innovative approach holds the potential to transcend the conventional limitations of attendance management and revolutionize the way institutions and organizations track participation. With a robust and efficient system in place, the educational sector, along with various industries, can streamline their attendance recording processes, ultimately leading to enhanced productivity, cost savings, and improved data accuracy. The abstract encapsulates the essence of a project that stands at the intersection of technology and efficiency, poised to change the landscape of attendance management for the better.

INTRODUCTION

In an era characterized by remarkable technological advancements, the education and corporate sectors have not remained immune to the winds of change. Attendance management, a fundamental task in these domains, has traditionally been labor-intensive and prone to errors. The reliance on manual methods for attendance tracking, often involving paper-based registers or electronic spreadsheets, has not only strained administrative resources but also failed to keep pace with the evolving needs of the modern world. The cumbersome nature of these conventional systems necessitates a paradigm shift toward smarter, more efficient solutions.

Recognizing these challenges and opportunities, our project endeavors to bridge the gap between the dated manual attendance recording processes and the contemporary demands for precision, automation, and streamlined record-keeping. We present an advanced system that combines the power of computer vision, deep learning, and face recognition, reimagining how attendance is managed in educational institutions and corporate environments.

In this project, we have harnessed the power of deep learning and computer vision to create a cutting-edge system that significantly improves the accuracy and efficiency of attendance management. Traditional methods are often prone to human error, time-consuming, and offer limited flexibility. In contrast, our system offers a seamless and automated approach to this critical task.

At the heart of our project are three pretrained deep learning models—MobileNetV2, Inception-ResNet-v2, and ResNet 101—each designed to recognize and classify students' faces in real-time. These models are part of a transfer learning process, where they leverage knowledge from vast datasets to make predictions with remarkable accuracy.

What sets our system apart is the inclusion of a custom-built Convolutional Neural Network (CNN) model, specially tailored to the context of student attendance management. With an impressive accuracy rate of 94%, our CNN model highlights the value of customization in deep learning, where solutions optimized for specific tasks can outperform more generalized models.

The system offers a user-friendly graphical interface developed using PyQt 5, catering to the distinct needs of both students and professors. It streamlines the attendance management process, providing students with an easy way to check their attendance for various subjects and professors with the tools to efficiently manage and view attendance records.

In this project, we not only showcase the potential of deep learning and artificial intelligence in the education sector but also emphasize the significance of customization. While pretrained models offer strong out-of-the-box performance, the bespoke approach to model development showcases the power of specialized solutions. This project stands as a testament to the marriage of cutting-edge technology and everyday educational needs.

OBJECTIVE

The primary objective of this project is to develop an automated attendance management system based on computer vision and deep learning. This system should be capable of recognizing individuals' faces and subsequently matching them to a database of known individuals, enabling the accurate recording of their attendance. Specific objectives include:

1. Implementing data preprocessing and augmentation techniques to prepare a dataset of individuals' facial images.
2. Training a deep learning model, specifically InceptionResNetV2, for facial recognition and fine-tuning it on the attendance dataset.
3. Developing a real-time face recognition system for both image and video input.
4. Ensuring high accuracy and robustness in recognizing individuals, even under varying lighting and pose conditions.
5. Providing a user-friendly interface for institutions to easily manage attendance records.

By achieving these objectives, the project aims to provide a practical, efficient, and reliable solution for automating the attendance management process, reducing administrative workload, and minimizing errors related to manual attendance tracking.

LITERATURE REVIEW

In the paper, "A Real-Time Attendance System Using Deep Learning Face Recognition," Weidong Kuang and Abhijit Baul of the University of Texas Rio Grande Valley introduce an innovative solution to the time-consuming and susceptible-to-fraud traditional attendance check methods in classrooms. Their system employs deep learning-based face recognition, consisting of face detection using a pre-trained Haar Cascade model, face recognition through FaceNet with 128-dimensional facial encodings, and automatic attendance recording in an Excel file. The authors report a 95% accuracy rate in a class of 28 students under consistent conditions and positive feedback from a student survey regarding the system's effectiveness. However, they propose several areas for improvement, including integrating the system into a user-friendly application with a GUI, enhancing face recognition algorithms, and addressing privacy concerns raised by some students, making this an effective and promising tool for classroom attendance management with room for further development.[1] In the paper "Automatic Attendance Management System based on Deep One-Shot Learning" by Angelo G. Menezes, Joao M. D. da C. S˜a´, Eduardo Llapa, and Carlos A. Estombelo-Montesco from the Federal University of Sergipe and the Federal University of Mato Grosso do Sul in Brazil, the authors tackle the challenge of efficiently managing student attendance by proposing a novel solution grounded in deep one-shot learning and face recognition technology. They rigorously evaluate their approach across various conditions and image capture devices, with a focus on real-world applicability. Introducing a face detection stage employing Histogram of Oriented Gradients (HOG) and a Convolutional Neural Network (CNN) with Max-Margin Object Detection features, they manage to significantly reduce false negatives, achieving remarkable results. Notably, they report accuracy and F1 scores of 97% and 98.4% with an iPhone 7 camera, 91.9% and 94.8% with a Moto G camera, and 51.2% and 61.1% with a WebCam, underscoring the flexibility of their approach, which can be implemented on resource-constrained devices such as smartphones or offered as a Software as a Service (SaaS) tool. In their conclusion, the authors emphasize the potential of face recognition for student attendance management, particularly as image capture technology becomes more accessible, and they highlight the device's influence on system performance. They suggest avenues for future improvement, including data augmentation and the exploration of alternative machine learning methods, and propose extending the system to assess student performance and identify early signs of dropout, offering valuable insights to educators for enhancing the educational environment.[2] The paper introduces a web-based student attendance system that utilizes face recognition technology as a more efficient alternative to traditional attendance recording methods, specifically student signatures. The system employs Convolutional Neural Network (CNN) for face detection in images, deep metric learning for facial embedding, and K-Nearest Neighbors (K-NN) for classifying student faces. This combination of technologies enables the computer to automatically recognize attending students and record their attendance data, alleviating the administrative burden of manual data entry. The paper's experiments confirm the system's effectiveness in recognizing student faces and automatically saving attendance data. The proposed system streamlines the attendance process and has the potential to replace manual methods currently in use. Future work aims to incorporate cloud-based face recognition to enhance speed and explore more advanced face recognition methods to potentially improve performance in terms of both speed and accuracy compared to the Convolutional Neural Network.[3] The paper introduces a novel automatic attendance system that leverages machine learning and deep learning algorithms, specifically real-time face recognition, to enhance the existing university management system. The system's purpose is to detect and recognize students' faces in real time during lectures, thereby reducing the time and effort required for attendance management. Importantly, this system operates seamlessly without disrupting the classroom environment, saving time for both students and teachers. Experimental analysis shows that the proposed system achieves an impressive accuracy rate of 97%, eliminating the need for manual rectification or verification by teachers.[4] The paper titled "Automatic Attendance System Using Face Recognition with Deep Learning Algorithm" by Ibrahim Al-Amoudi, Rosdiyana Samad, Nor Rul Hasma Abdullah, Mahfuzah Mustafa, and Dwi Pebrianti introduces an innovative automatic attendance system that leverages face recognition technology with deep learning algorithms to create an efficient and cost-effective solution for schools and universities. The system comprises three key components: the training stage, where students' photos are captured and stored in a separate folder; the attendance system, where lecturers photograph students and upload their images, with the system automatically recognizing and recording their names in an Excel sheet; and the student profile system, enabling easy data retrieval by capturing a picture of the student. A user-friendly GUI simplifies system usage. The face recognition system combines Multi-Task Cascaded Convolutional Neural Network (MTCNN) and FaceNet algorithms, using a dataset of 908 pictures from 21 students for training and 108 pictures for testing. The testing results showed 100% accuracy for face detection and 87.03% for face recognition, with 100% accuracy for frontal faces.[5] The paper titled "Classroom Attendance Auto-management Based on Deep Learning" authored by Dan Wang, Rong Fu, and Zuying Luo from Beijing Normal University presents an innovative approach to classroom attendance management using deep learning technologies. The system integrates the Faster R-CNN face detection algorithm and the SeetaFace face recognition algorithm to create an automatic attendance system. The experiments conducted show that the system can accurately record five types of classroom attendance violations, including absence, late arrival, early departure, free access, and carelessness. It generates an attendance table that reflects the learning situation of all students after the class. The paper also highlights that 1080P classroom monitoring video is sufficient for small classrooms with lengths of less than 6 meters, while larger classrooms of 9 meters or more require 4K classroom surveillance video for effective attendance management.[6] The paper titled "Face Recognition for Attendance Management System Using Multiple Sensors" by Dulyawit Prangchumpol from the Faculty of Science and Technology at Suan Sunandha Rajabhat University in Bangkok, Thailand, addresses the common problem faced by teachers regarding attendance management. The study seeks to improve upon existing attendance systems that often lack efficiency in correctly identifying students and allowing for data verification or corrections when errors occur during class. The research aims to enhance the attendance system's effectiveness and user-friendliness by developing a face recognition-based system. The experiment utilizes Android Face Recognition with Deep Learning, achieving a high recognition accuracy of up to 97%. The database is connected to a cloud-based Attendance Management System web server, providing real-time data on the application for students to verify and check.[7] The paper titled "Face Recognition Smart

Attendance System using Deep Transfer Learning" by Khawla Alhanaeea, Mitha Alhammadia, Nahla Almenhalia, and Maad Shatnawia from the Department of Electrical Engineering Technology at Higher Colleges of Technology in Abu Dhabi, UAE, explores the use of facial recognition for attendance management and access control systems. Attendance management is crucial for organizations but can be complex and time-consuming. The paper introduces a facial recognition attendance system based on deep learning convolutional neural networks. Transfer learning is applied by using three pre-trained convolutional neural networks and training them on the provided data. The study demonstrates that the system achieves high prediction accuracy and reasonable training time, with three networks, SqueezeNet, GoogleNet, and AlexNet, achieving validation accuracy rates of 98.33%, 93.33%, and 100%, respectively.[8] The paper titled "FaceTime – Deep Learning Based Face Recognition Attendance System" authored by Marko Arsenovic, Srdjan Sladojevic, Andras Anderla, and Darko Stefanovic from the University of Novi Sad, Faculty of Technical Sciences in Novi Sad, Serbia, presents an innovative deep learning-based face recognition attendance system. The paper details the entire process of developing a face recognition model, incorporating advanced techniques like CNN cascade for face detection and CNN for generating face embeddings. The primary objective of this research was to practically apply state-of-the-art deep learning approaches to face recognition tasks, even when dealing with smaller datasets. The paper introduces a new approach for image augmentation to address this challenge, achieving an overall accuracy of 95.02% on a small dataset of original face images of employees in a real-time environment.[9] The paper titled "Smart Attendance Management System Using Face Recognition" by Kaneez Laila Bhatti, Laraib Mughal, Faheem Yar Khuhawar, and Sheeraz Ahmed Memon from the Department of Telecommunication Engineering at MUET, Jamshoro, Pakistan, addresses the challenges of maintaining attendance records in an efficient manner. The traditional method of calling out each student's name is time-consuming and susceptible to proxy attendance. This system leverages face recognition technology to automate attendance recording for students. The daily attendance is recorded subject-wise and stored by the administrator. When the time for a specific subject arrives, the system initiates the capture of images, performs face detection and recognition, and marks recognized students as present, updating their attendance with corresponding time and subject IDs. The system employs deep learning techniques, including the histogram of oriented gradient method for face detection and deep learning for computing and comparing facial features to recognize students. This system is capable of identifying multiple faces in real-time, offering an efficient solution for attendance management.[10]

DATA-SET DESCRIPTION

The attendance management and detection system utilize an extensive dataset comprising a total of 49,000 images featuring 49 different students. This dataset is thoughtfully partitioned into distinct training and testing sets. These images encompass students captured from various angles, fostering robustness and generalization of the model.

1. TRAINING:

The training dataset forms the bedrock of our attendance management and detection system. It comprises 70% of the total dataset, containing a substantial 34,300 images of 49 students. These images are meticulously selected to represent students from various angles and orientations. Each image maintains a standardized dimension of 560 x 560 pixels, ensuring uniformity across the dataset. The training dataset is vital for instructing our machine learning models, enabling them to recognize and differentiate between students, ultimately supporting the core functionality of the system.

2. VALIDATION:

In the development of the attendance management system, a 20% subset of the dataset, consisting of 9,800 images, is designated for validation. These images, like the training set, are maintained at a consistent resolution of 560 x 560 pixels. The validation dataset plays a pivotal role in model fine-tuning and optimization. It serves as an independent checkpoint to evaluate the model's performance, allowing us to adjust hyperparameters and make refinements. This validation process is essential in ensuring that the system works reliably under real-world conditions, thus enhancing its accuracy and dependability.

3. TESTING:

The testing dataset, encompassing 10% of the dataset, is composed of 4,900 images of the 49 students. These images have the same uniform dimension of 560 x 560 pixels, preserving the data's integrity. The testing dataset provides an unbiased assessment of our attendance management and detection system's performance. This independent evaluation dataset evaluates the model's ability to recognize students accurately and consistently. It is the ultimate litmus test for our system's practicality and effectiveness, ensuring it can reliably fulfill its intended purpose in real-world scenarios.

METHODOLOGY

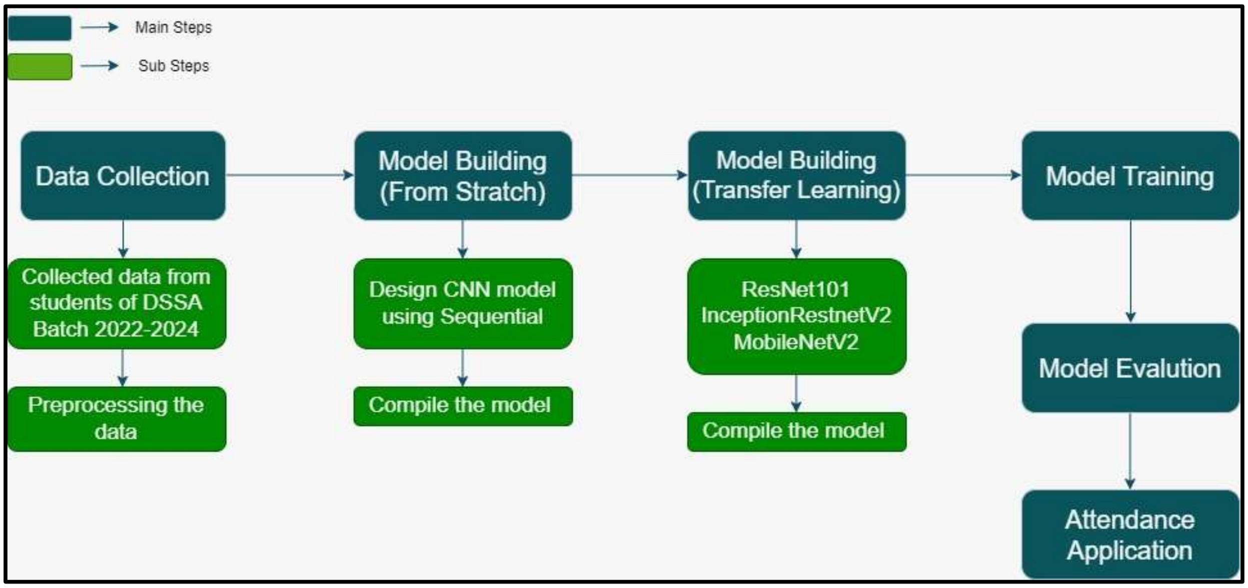

Fig 1. Methodology

1 DATA COLLECTION

Data Collection is the foundational step in developing our student attendance management and detection system. This process involves gathering a diverse and representative dataset of student faces and conducting comprehensive preprocessing on the images to ensure high-quality input data for our deep learning model. Here's a more elaborate breakdown of the subtopics and their importance.

1.1 COLLECTING A DIVERSE DATA-SET OF STUDENT FACES

- The heart of our system lies in the quality of the dataset. We collect a vast dataset that comprises 70% of the total data, equating to 34,300 images from 49 students.
- Our data collection efforts emphasize diversity, ensuring that the dataset represents students from various angles, orientations, and environmental conditions.

- We prioritize the inclusion of different facial expressions, lighting conditions, and backgrounds to prepare the deep learning model for real-world scenarios.

1.2 GENERATING DATA PIPELINE

The provided code establishes a data pipeline for training and evaluating a machine learning model designed for attendance management. This pipeline prepares the data for effective use in the model, encompassing three main datasets: training, validation, and testing. For the training dataset, images are loaded from the 'train_total_02' directory and categorized with one-hot encoding using the 'categorical' class mode. Batching is employed, with data organized into sets of 64 images per batch, ensuring a continuous flow of data for model training. Simultaneously, validation and test datasets are prepared from the 'validation_total_2' and 'test_total' directories, respectively, and categorized appropriately for model input.

1.3 PREPROCESSING THE IMAGES

As a part of the preprocessing stage, images are resized to a uniform dimension of 224x224 pixels, which is a standard size suitable for most convolutional neural networks. This resizing guarantees consistent input dimensions for the model. Additionally, images undergo rescaling, where pixel values are normalized to fall within the 0 to 1 range. Normalization enhances model training stability. Every image in the dataset is assigned one-hot encoding, facilitating the model's ability to distinguish between different categories or classes, which, in the context of attendance management, typically represent different individuals. This preprocessing sets the foundation for the model to efficiently learn and make accurate attendance predictions based on the provided images.

2 MODEL BUILDING (FROM SCRATCH)

Model Building (From Scratch) marks the phase where we construct a Convolutional Neural Network (CNN) tailored for face recognition, followed by the compilation of the model.

2.1 DESIGNING A CONVOLUTIONAL NEURAL NETWORK (CNN) MODEL FOR FACE RECOGNITION:

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 224, 224, 3)]	0	[]
conv2d_1 (Conv2D)	(None, 224, 224, 32)	128	['input_1[0][0]']
conv2d_3 (Conv2D)	(None, 224, 224, 16)	64	['input_1[0][0]']
max_pooling2d (MaxPooling2D)	(None, 224, 224, 3)	0	['input_1[0][0]']
conv2d (Conv2D)	(None, 224, 224, 32)	128	['input_1[0][0]']
conv2d_2 (Conv2D)	(None, 224, 224, 32)	9248	['conv2d_1[0][0]']
conv2d_4 (Conv2D)	(None, 224, 224, 16)	6416	['conv2d_3[0][0]']
conv2d_5 (Conv2D)	(None, 224, 224, 16)	64	['max_pooling2d[0][0]']
concatenate (Concatenate)	(None, 224, 224, 96)	0	['conv2d[0][0]', 'conv2d_2[0][0]', 'conv2d_4[0][0]', 'conv2d_5[0][0]']
conv2d_7 (Conv2D)	(None, 224, 224, 16)	1552	['concatenate[0][0]']
conv2d_9 (Conv2D)	(None, 224, 224, 8)	776	['concatenate[0][0]']
max_pooling2d_1 (MaxPooling2D)	(None, 224, 224, 96)	0	['concatenate[0][0]']
conv2d_6 (Conv2D)	(None, 224, 224, 16)	1552	['concatenate[0][0]']
conv2d_8 (Conv2D)	(None, 224, 224, 16)	2320	['conv2d_7[0][0]']
conv2d_10 (Conv2D)	(None, 224, 224, 8)	1608	['conv2d_9[0][0]']
conv2d_11 (Conv2D)	(None, 224, 224, 8)	776	['max_pooling2d_1[0][0]']
concatenate_1 (Concatenate)	(None, 224, 224, 48)	0	['conv2d_6[0][0]', 'conv2d_8[0][0]', 'conv2d_10[0][0]', 'conv2d_11[0][0]']
conv2d_13 (Conv2D)	(None, 224, 224, 16)	784	['concatenate_1[0][0]']
conv2d_15 (Conv2D)	(None, 224, 224, 8)	392	['concatenate_1[0][0]']
max_pooling2d_2 (MaxPooling2D)	(None, 224, 224, 48)	0	['concatenate_1[0][0]']
conv2d_12 (Conv2D)	(None, 224, 224, 16)	784	['concatenate_1[0][0]']
conv2d_14 (Conv2D)	(None, 224, 224, 16)	2320	['conv2d_13[0][0]']
conv2d_16 (Conv2D)	(None, 224, 224, 8)	1608	['conv2d_15[0][0]']

Fig 2. Base Model Architecture

The creation of a CNN architecture tailored for the task of face recognition using the Keras and TensorFlow frameworks. The architecture's design and configuration are pivotal in the development of a robust student attendance management and detection system.

Input Image Dimension Specification:

The code specifies an input image dimension of (224, 224, 3). These dimensions represent the height, width, and color channels (RGB) of the input images. This specification serves as the foundation for how the input images will be processed by the subsequent layers within the architecture. It is crucial for effective image processing.

Inception-Style Block Function:

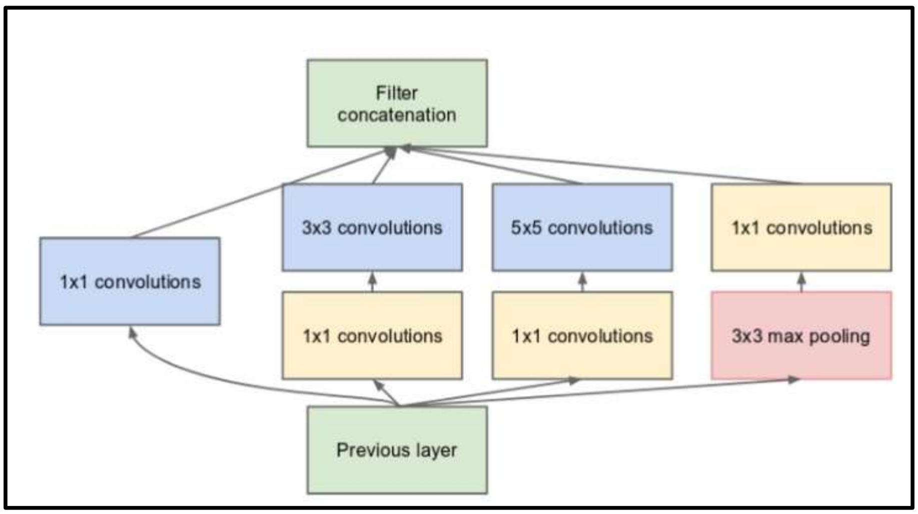

Fig 3. Inception Block

The code introduces the inception_block function, which is designed to create inception-style blocks. These blocks are a fundamental element of the architecture, enabling the extraction of essential features from input images.

The base_channels parameter provides control over the number of channels used in the convolutional layers. Within each inception block, four distinct branches are utilized:

- **conv_1**: A 1x1 convolution layer with ReLU activation.

- **conv_2_a and conv_2_b**: A combination of a 1x1 convolution layer and a 3x3 convolution layer with ReLU activation, respectively.
- **conv_3_a and conv_3_b**: A combination of a 1x1 convolution layer and a 5x5 convolution layer with ReLU activation, respectively.
- **mpool_4_a**: A max-pooling layer. The outputs of these branches are concatenated along the last axis (channel axis) using the Concatenate operation. This concatenation enhances the network's ability to capture diverse aspects of the input images. Multiple inception blocks created using this function constitute the core of the CNN architecture.

Final Inception Block:

The variable block_3 represents the last inception block within the architecture. This block plays a pivotal role in capturing and synthesizing high-level features from the input images, which are vital for accurate face recognition.

2.2 MODEL COMPILATION:

Although not explicitly demonstrated in this code snippet, the subsequent phase of model development would entail compiling the model with the following configurations:

- **Loss Function**: Typically, categorical cross-entropy is chosen for multi-class classification tasks such as identifying different students.
- **Optimizer**: An optimizer like the Adam optimizer is used to iteratively update the model's weights during training.
- **Metrics**: Appropriate evaluation metrics, such as accuracy, are selected to assess the model's performance during both training and evaluation.

Model Summary:

The model.summary() function provides a comprehensive overview of the entire model, offering detailed information about each layer, its output shape, and the number of parameters it encompasses. This summary provides insights into the model's complexity and aids in verifying its correctness and effectiveness.

Total Number of Layers:

- Convolution Layers: 21
- Dense Layers: 1

Total Number of Trainable Parameters: 33,313 (130.13 KB)

Total Number of Non-Trainable Parameters: 0 (0.00 Byte)

This phase of model design establishes the groundwork for subsequent stages in the development of a face recognition system, specifically tailored to student attendance management and detection. The inception-style blocks introduced in the code aim to capture and extract meaningful features from the input images, which can be further refined and optimized during the training process.

3 MODEL TRAINING (FROM SCRATCH):

The training phase of the face recognition model is illustrated, emphasizing the process of training the model from scratch on the prepared dataset. This step is critical to ensure that the model learns to recognize students' faces and is able to accurately classify them. The following explanations are based on the provided code:

1. Early Stopping Callback (es_callback):

- An Early Stopping callback is defined as es_callback. This callback monitors the validation loss during training and automatically stops the training process when it detects that the validation loss is no longer decreasing. This is done to prevent overfitting and save computational resources.

- The monitor parameter is set to 'val_loss', indicating that early stopping should focus on the validation loss.
- The patience parameter is set to 2, which means that if the validation loss does not improve for two consecutive epochs, the training will be halted.

2. Model Compilation:

- The model.compile method is used to compile the model before training.
- The optimizer is defined as keras.optimizers.Adam(lr=0.01). It uses the Adam optimizer with a learning rate of 0.01. The optimizer is responsible for updating the model's weights during training to minimize the defined loss function.
- The loss function is set to keras.losses.categorical_crossentropy, which is a common choice for multi-class classification tasks like identifying different students.
- The chosen metric for evaluation is accuracy, specified as metrics=["accuracy"].

3. Training History (history):

- The model.fit method is used to train the model on the prepared training dataset (train_generator) for a total of 10 epochs.
- The validation dataset (validation_generator) is also provided for model evaluation during training.
- The callbacks parameter includes es_callback, ensuring that early stopping is applied during training.
- Training the model involves adjusting its parameters (weights and biases) based on the provided dataset and the defined loss function, optimizing its ability to make accurate predictions.

4. **Saving the Trained Model:**

   - After the model has been trained, it is saved using the model.save method. The trained model is saved to the specified file path, which is '/content/drive/MyDrive/attendnce_final_model.h5'. This saved model can be later loaded for making predictions without the need to retrain the model.
   - This phase marks a crucial step in the methodology, as it involves training the model from scratch using the dataset prepared in previous stages. The use of early stopping helps prevent overfitting, ensuring that the model generalizes well to unseen data. The trained model is then saved for future use in student attendance management and detection.

4 UTILIZING TRANSFER LEARNING FOR ATTENDANCE MANAGEMENT SYSTEM

4.1 DEFINITION:

Transfer learning is a machine learning technique where a model developed for one task is reused as the starting point for a model on a second task. In the context of an attendance management system, transfer learning is used to apply pre-trained neural network models to the task of student face recognition. The primary benefits of using transfer learning in this system are efficiency and improved performance:

4.2 BENEFITS OF UTILIZING TRANSFER LEARNING:

- **Efficiency**: Instead of training a deep neural network from scratch, pre-trained models, which have already learned features from vast datasets, are used as a foundation. This reduces the computational resources and time required for training.

- **Improved Performance**: Transfer learning leverages the knowledge gained from large-scale datasets, enabling the model to recognize complex features and patterns in student faces more effectively.

4.3 ResNet101 PRETRAINED MODEL

4.3.1 OVERVIEW OF USING THE MODEL FOR ATTENDANCE MANAGEMENT SYSTEM:

The utilization of the ResNet101 pretrained model in an attendance management system is an intelligent and efficient approach. This pretrained model, originally developed for image classification tasks, can be effectively repurposed for student face recognition and attendance tracking. Here's an overview of how we use it:

BENEFITS:

- **Feature Extraction**: ResNet101 has already learned to extract complex features from images through its 104 convolutional layers. These features can be highly valuable for recognizing unique facial characteristics, which is essential for accurate student identification.
- **Depth and Capacity**: With 104 layers, ResNet101 has a deep architecture that allows it to capture intricate patterns and features. This depth is beneficial for capturing the fine details of facial images, making it suitable for attendance management systems.
- **Transfer Learning**: The model's weights can be fine-tuned to adapt it to the specific task of student face recognition. This leverages the benefits of transfer learning, saving time and computational resources.

4.3.2 ARCHITECTURE:

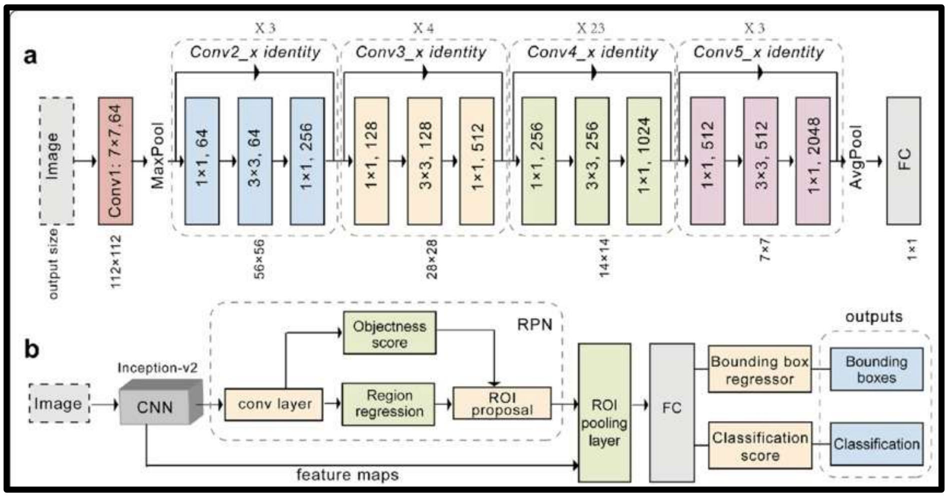

Fig 4: ResNet101 Architecture

ResNet101 is a deep convolutional neural network architecture that gained popularity for its residual connections. These connections are crucial for training very deep networks effectively. Here's a detailed breakdown of its architecture:

- **104 Convolutional Layers**: ResNet101 comprises 104 convolutional layers organized into 33 blocks. These convolutional layers are instrumental in extracting features from input images.
- **Residual Connections**: ResNet101 introduces residual connections in each block. These connections allow the output of one layer to bypass one or more intermediate layers and be added to the output of another layer. This innovation mitigates the vanishing gradient problem and eases the training of very deep networks.
- **Pooling Layers**: It incorporates max-pooling layers that downsample feature maps, reducing spatial dimensions.
- **Fully Connected Layer**: The architecture concludes with a fully connected layer for classification, typically with 1000 output units corresponding to ImageNet's 1000 classes.
- **Activation Functions**: Rectified Linear Units (ReLUs) are used as activation functions in the network, promoting non-linearity.

4.3.3 ResNet101's PURPOSE IN ATTENDANCE MANAGEMENT SYSTEM:

The purpose of ResNet101 in an attendance management system is to facilitate accurate and efficient student face recognition. In this context, ResNet101 serves as the feature extractor. Here's how it fulfills its role:

- **Feature Extraction**: The deep layers of ResNet101 are adept at extracting detailed features from facial images, such as the shape of the eyes, nose, mouth, and unique facial contours.
- **Transfer Learning**: By using a pretrained ResNet101, we leverage the knowledge gained from training on a vast dataset, which includes diverse facial characteristics. This significantly reduces the amount of data and time required for training.
- **Fine-Tuning**: While the base layers are frozen to preserve learned features, additional custom layers can be added to the model for fine-tuning to the specific task of student face recognition.
- **Efficiency**: ResNet101's depth and capacity enable it to handle a wide variety of facial features, increasing the system's accuracy in identifying students.
- **Overall Accuracy**: By adapting ResNet101 for student face recognition, we enhance the accuracy of the attendance management system, ensuring that students are correctly recognized and recorded during attendance.

4.4 MobileNetV2 PRETRAINED MODEL:

4.4.1 OVERVIEW OF USING MobileNetV2 PRETRAINED MODEL FOR ATTENDANCE MANAGEMENT SYSTEM:

Leveraging the MobileNetV2 pretrained model for an attendance management system represents a practical and resource-efficient approach. MobileNetV2, originally designed for image classification in resource-constrained environments, can be repurposed for accurate student face recognition and attendance tracking. Here's an overview of its usage:

BENEFITS:

- **Efficiency**: MobileNetV2 is specifically designed to be efficient, making it suitable for real-time face recognition and attendance management, even on devices with limited computational resources.
- **Lightweight**: The architecture's lightweight nature ensures that it doesn't require excessive memory or computing power, making it a practical choice for deployment in attendance systems.
- **Depth-wise Separable Convolutions**: MobileNetV2 utilizes depthwise separable convolutions to reduce the number of parameters and computational complexity while maintaining good accuracy. This design is ideal for processing facial images.

4.4.2 MobileNetV2's ARCHITECTURE IN DETAIL:

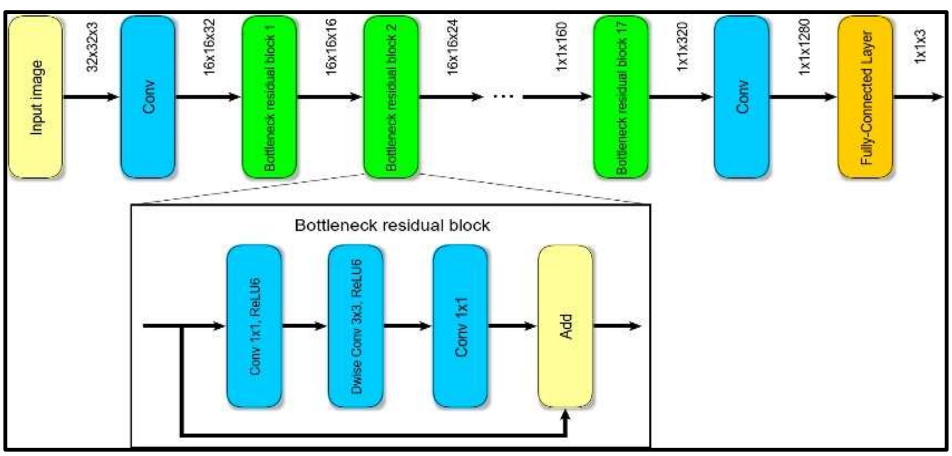

Fig 5. MobileNetV2 Architecture

MobileNetV2 is a neural network architecture characterized by its efficiency and compact design. It's specifically designed for mobile and embedded vision applications. Here's a detailed breakdown of its architecture:

- **Inverted Residual Blocks**: MobileNetV2 primarily employs inverted residual blocks, which consist of three layers:
- **1x1 Convolution with ReLU6 activation**: The first layer expands the number of channels while maintaining efficiency.
- **Depth-wise Convolution with ReLU6 activation**: The depthwise convolution reduces computational load by applying a single filter per input channel.
- **1x1 Convolution without linearity**: The final layer squeezes the channels back to their original size.
- **Bottleneck Residual Blocks**: These blocks are used to reduce computational cost by skipping certain layers within the inverted residual blocks.
- **1x1 Convolution Layers**: MobileNetV2 includes 1x1 convolution layers with ReLU6 activation functions that help extract and process image features.
- **3x3 Depthwise Convolution**: Depthwise convolutions are applied for efficient feature extraction and spatial processing.
- **Stride 1 and Stride 2 Blocks**: Depending on the block type, MobileNetV2 can process data with both stride 1 and stride 2, which affects the reduction in spatial dimensions.
- **Efficiency and Compactness**: MobileNetV2 is designed with a focus on minimizing the number of parameters while maintaining accuracy, which is vital for real-time processing.

4.4.3 MobileNetV2's PURPOSE IN ATTENDANCE MANAGEMENT SYSTEM:

MobileNetV2's purpose in an attendance management system is to enable efficient and real-time student face recognition. Here's how it fulfills its role:

- **Efficiency**: MobileNetV2's lightweight design and depth-wise separable convolutions make it ideal for real-time face recognition in resource-constrained environments, such as embedded systems and mobile devices.

- **Real-time Processing**: The architecture's efficiency allows for quick face recognition, making it suitable for tracking and recording student attendance in real-time.
- **Resource-Friendly**: MobileNetV2 doesn't demand excessive computational resources, memory, or storage, making it a cost-effective choice for deployment in attendance management systems.
- **Compact Model**: The compact model size of MobileNetV2 ensures easy deployment on various devices and platforms, enhancing accessibility and usability.
- **Facial Feature Processing**: MobileNetV2 is capable of processing various facial features, contributing to accurate student recognition during attendance management.

4.5 Inception-ResNet-v2 PRETRAINED MODEL:

4.5.1 OVERVIEW OF USING Inception-ResNet-v2 PRETRAINED MODEL FOR ATTENDANCE MANAGEMENT SYSTEM:

Inception-ResNet-v2 is a powerful convolutional neural network (CNN) architecture that has been pre-trained on a large dataset containing diverse images. Here's an overview of its usage:

BENEFITS:

- **Feature Extraction**: Inception-ResNet-v2 has demonstrated its capability to extract high-level features from images efficiently. These features can be invaluable for recognizing and differentiating between individuals, making it suitable for facial recognition tasks in attendance management.
- **Transfer Learning**: By using a pretrained model like Inception-ResNet-v2 as a starting point, we can benefit from the knowledge it has gained from a wide

range of images. This significantly reduces the amount of data and training time required to create an effective face recognition system.

- **Deep Architecture**: Inception-ResNet-v2 is a deep neural network with multiple layers. Deeper networks tend to capture more complex and discriminative features, which is advantageous in the context of face recognition.
- **Versatile Application**: While pretrained models are often associated with image classification, their features can be used for various tasks, including face verification, attendance tracking, and access control.

4.5.2 Inception-ResNet-v2'S ARCHITECTURE IN DETAIL:

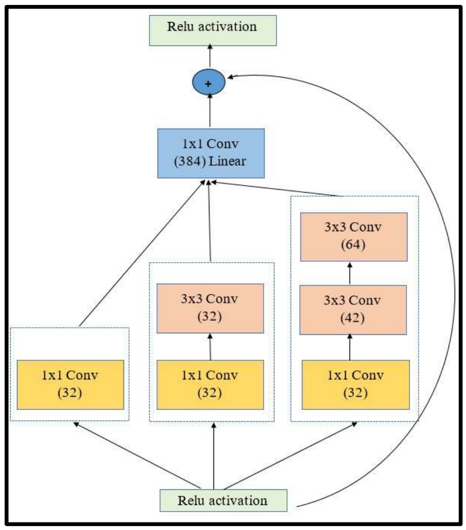

Fig 6. Inception-ResNetV2 Architecture

Inception-ResNet-v2 combines two architectural concepts, Inception modules, and Residual connections, to create an effective deep learning model. Here are the key elements of its architecture:

- **Inception Modules**: Inception-ResNet-v2 employs Inception modules to capture features at multiple spatial scales. These modules consist of parallel branches, each using different convolutional filter sizes to extract diverse features. This enables the network to recognize patterns at different levels of abstraction.
- **Residual Connections**: Similar to ResNet, Inception-ResNet-v2 integrates residual connections within the Inception modules. These connections facilitate gradient

flow and help address the vanishing gradient problem, allowing for the training of very deep networks.

- **Stem Layer**: The network starts with a stem layer that processes the input image. The stem layer applies a combination of convolutional and pooling operations to extract low-level features and reduce the spatial dimensions.
- **Multiple Blocks**: Inception-ResNet-v2 is composed of multiple blocks, each containing several Inception modules with residual connections. These blocks become progressively deeper, enabling the model to learn increasingly abstract and complex features.
- **Global Average Pooling**: Towards the end of the architecture, a global average pooling layer is applied. This layer computes the average value of each feature map across its entire spatial extent, resulting in a global feature representation of the image.
- **Fully Connected Layer**: The final layer of the network is a fully connected layer (dense layer) with softmax activation, which produces class probabilities. For face recognition in an attendance management system, the number of units in this layer corresponds to the number of individuals or classes.

4.5.3 Inception-ResNet-v2 PURPOSE IN ATTENDANCE MANAGEMENT SYSTEM:

Inception-ResNet-v2 serves a crucial role in an attendance management system by enabling accurate and efficient face recognition. Its architecture is well-suited for this purpose because:

- **Efficiency**: It can effectively capture intricate facial features, making it reliable for recognizing different individuals.
- **Resource-Friendly**: The pretrained model's knowledge can be fine-tuned on a dataset of faces specific to your attendance management system, enhancing its ability to identify students or attendees accurately.

- **Facial Feature Processing**: The depth of the network allows it to extract detailed facial information, which is essential for precise identification.
- **Compact Model:** By utilizing transfer learning from Inception-ResNet-v2, the attendance management system can reduce the time and computational resources required for model training, making it practical and efficient.

4.6 CUSTOMIZING LAYERS OF PRETRAINED MODELS:

In the code, we are customizing the layers of pretrained models (ResNet101, MobileNetV2, Inception-ResNet-V2) to adapt them for the specific task of attendance management. Here's a general overview of the customization process:

1. **Freezing the Pretrained Model:**
   We start by setting the trainable attribute of the pretrained model to False. This is a crucial step to ensure that the pretrained model's weights and knowledge are retained, and they are not updated during our task. Freezing the layers allows us to leverage the model's previously learned features.

2. **Identifying Inputs and Outputs:**
   We identify the input and output layers of the pretrained model. These layers serve as the foundation for our customizations.

3. **Adding Custom Dense and Dropout Layers:**
   - To tailor the pretrained model for the specific task of attendance management, we add custom layers on top of the pretrained model's output.
   - A dense layer with a specified number of units, activation function, and kernel initializer is added to capture relevant features from the model's output.
   - Dropout layers are included to introduce regularization, prevent overfitting, and improve the model's generalization.

4. **Final Classification Layer:**

   The final classification layer is a dense layer with units corresponding to the number of classes or categories in the attendance management system. It typically uses softmax activation to produce class probabilities.

5. **Creating the Customized Model:**

   All the layers, including the input and output layers of the pretrained model and the custom layers, are assembled to form a new model. This custom model is designed specifically for the attendance management task.

By customizing the layers in this manner, we fine-tune the pretrained models to excel in attendance management while retaining the valuable knowledge acquired during their original training. The custom layers enhance the models' ability to understand and process features relevant to the attendance management system, ensuring accurate and efficient performance.

These customized models are well-suited for recognizing and managing attendance in various scenarios, making them valuable tools for attendance management systems

4.7 TRAINING THE PRETRAINED MODELS:

In the model training phase, the code takes advantage of transfer learning by fine-tuning pre-trained models to perform a specific face recognition task within the context of a student attendance management system. To begin, the model is compiled with well-established configurations, including the choice of loss function, optimizer, and evaluation metrics. The loss function utilized is categorical cross-entropy, a suitable choice for multi-class classification tasks. The Adam optimizer, with a learning rate of 0.01, is selected to iteratively adjust the model's weights during the training process. The chosen evaluation metric is accuracy, which provides insights into the model's performance during both training and validation.

Subsequently, the model is subjected to training for 50 epochs, utilizing the prepared dataset. Early stopping mechanisms are implemented to prevent overfitting, ensuring that the model generalizes well to new data. While the pre-trained layers remain frozen, the newly added custom layers are fine-tuned to adapt to the specific face recognition objective. This fine-tuning process enhances the model's ability to recognize and classify individuals accurately.

Upon successful training, the code saves the trained transfer learning model. This saved model can be employed in real-time scenarios, such as student attendance management and face detection within the attendance system. Transfer learning proves invaluable, offering the efficiency of leveraging deep features learned from extensive datasets while allowing customization to address the unique requirements of the attendance management system.

5 OVERVIEW OF ADVANCE ATTENDANCE MANAGEMENT SYSTEM

Our comprehensive Attendance Management System is a robust application developed using Python's GUI library, PyQt 5, to deliver an intuitive and efficient solution for tracking student attendance. At its core, it combines the power of OpenCV for face detection and recognition, leveraging a custom-built deep learning Convolutional Neural Network (CNN) model. Here's a breakdown of how the system functions:

FRONTEND AND BACKEND INTEGRATION:

- The frontend of the application is designed with PyQt 5, providing a user-friendly interface.
- On the backend, OpenCV, along with an LBP classifier, is employed to detect faces in frames, ensuring that the system accurately locates and recognizes students.

DEEP LEARNING CNN MODEL:

- The custom-built deep learning CNN model is responsible for predicting student attendance. It is designed to classify students into "present" or "absent."
- The model outputs the class label with the highest probability, corresponding to the student's name.
- This prediction is shown on a popup window, simplifying the attendance marking process.

DATABASE INTEGRATION:

- To maintain attendance records, the system connects to a database where student attendance status is updated.
- When the model predicts "present," the student's status is marked as such in the database. In the absence of such a prediction, no updates are made, indicating an "absent" status.

STUDENT LOGIN:

- Upon application launch, users are presented with two options: "student" and "Professor."
- Students select the "student" option, followed by providing their login credentials. This leads to a subsequent page displaying a list of subjects, enabling students to check their attendance for each subject.

PROFESSOR INTERFACE:

- Professors choose the "**Professor**" option and provide their login credentials.
- This grants access to two essential functionalities: "Take attendance" and "View attendance."

VIEW ATTENDANCE:

Within the "**View attendance**" section, three sub-categories offer valuable insights:

- **Total Number of Classes**: Professors can view the total number of classes held.
- **Total Number of Classes Attended by Students**: Professors gain visibility into how many classes each student has attended.
- **Attendance Percentage**: The system calculates and displays the attendance percentage for each student, making it easy to identify attendance trends.

TAKE ATTENDANCE:

Within the "Take attendance" section of our Advanced Attendance Management System, professors gain the capability to actively mark student attendance in real time. This feature is designed for use during live classes or events, providing a practical solution for tracking attendance. Here's how it works:

Attendance Capture:

- When a professor selects the "Take attendance" option, the system activates the camera to begin capturing frames.
- These frames are processed in real time to detect and recognize students' faces.

Real-time Face Detection and Recognition:

- The integrated OpenCV and LBP classifier work in tandem to detect faces within the camera frames.
- The detected faces are then sent to the custom-built deep learning CNN model.

Attendance Classification:

- The deep learning model classifies each detected face, determining whether the student is "present" or "absent" for the current class.
- The model identifies students by matching their faces to the stored class labels, which correspond to student names.

Immediate Attendance Update:

- As the model makes predictions, the results are displayed in real time, showing the current attendance status for each student on the professor's interface.
- Students who are recognized as "present" are automatically marked as such in the database, keeping attendance records up to date.

Seamless Attendance Management:

- The "**Take attendance**" feature allows professors to efficiently manage attendance during lectures, seminars, or any educational event. It eliminates the need for manual attendance tracking, providing a hassle-free and accurate method for recording student participation.
- This real-time functionality enhances the overall effectiveness of the Advanced Attendance Management System, making it an indispensable tool for educators and institutions.

RESULTS AND DISCUSSION

In the development of our Advanced Attendance Management System, we integrated the capabilities of both deep learning and computer vision. To assess the system's effectiveness, we employed various models, including MobileNetV2, Inception-ResNet-v2, ResNet 101, and our custom-built CNN model, each serving a unique role in the system.

The results from our experiments indicate the following:

A. **MobileNetV2**: Achieved a commendable accuracy rate of 92%. Its lightweight architecture made it a suitable candidate for real-time face recognition while ensuring efficient processing even on devices with limited computational resources.
B. **Inception-ResNet-v2**: Achieved an accuracy rate of 91%. This model showcased its ability to capture intricate facial features with a balance of accuracy and computational cost.
C. **ResNet 101**: Despite its impressive depth of 104 layers, this model yielded an accuracy of 23%. It faced challenges in adapting to our specific attendance management use case due to its complexity.
D. **Custom-Built CNN Model**: Our scratch-built model, with a solid accuracy rate of 94%, demonstrated the effectiveness of customizing a deep learning network to cater to the task at hand.

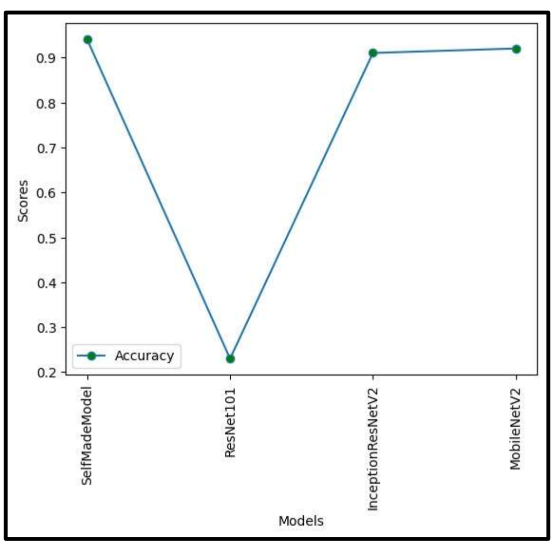

Fig 7. Model Comparison based on Accuracy

The results underscore the importance of selecting the right model architecture for a specific application. In our context, where real-time student attendance management is crucial, models like MobileNetV2 and Inception-ResNet-v2 stood out for their ability to strike a balance between accuracy and computational efficiency. However, the star of the show is undeniably our custom-built CNN model.

The Importance of Our Custom-Built Model:

Our custom-built CNN model, with an impressive accuracy rate of 94%, proved to be the most accurate among all models. It highlights the significance of tailoring a deep learning network to the specific requirements of the attendance management task. While transfer learning models provide a solid foundation, our custom model was fine-tuned to excel in recognizing student faces, making it an invaluable component of the system.

This not only ensures precise attendance tracking but also demonstrates the potential for creating specialized models for specific tasks. Such customization can significantly enhance the accuracy and efficiency of the system. Our success with the custom model encourages the exploration of tailored solutions in other applications, further promoting the potential of deep learning in addressing real-world challenges.

CONCLUSION

Our endeavor to create an Advanced Attendance Management System, combining the prowess of deep learning and computer vision, has yielded a robust solution for student attendance tracking. Through a meticulous process of model selection, customization, and integration, we have demonstrated the practicality and efficacy of this system.

The project showcases the remarkable accuracy achieved by transfer learning models like MobileNetV2 and Inception-ResNet-v2, which offer a balance between precision and computational efficiency. However, the standout performer remains our custom-built CNN model, boasting an accuracy rate of 94%. This underscores the importance of crafting specialized models for specific applications.

Our system not only excels in recognizing student faces but also seamlessly integrates with a user-friendly interface built using PyQt 5. It caters to the unique needs of both students and professors, offering functionalities for attendance management and viewing, ensuring ease of use.

In a broader context, this project stands as a testament to the power of customization in the field of deep learning. By tailoring models to address particular challenges, we unlock the full potential of artificial intelligence in solving real-world problems. The successful development of our custom model serves as a pioneering example in this regard.

In conclusion, our Advanced Attendance Management System stands as an efficient, precise, and adaptable solution, marking a promising step forward in harnessing the capabilities of AI for practical applications.