Implementation of ML model for image classification

A Project Report

submitted in partial fulfillment of the requirements

of

AICTE Internship on AI: Transformative Learning
with
TechSaksham – A joint CSR initiative of Microsoft & SAP

by

Arkadip Ghosh

www.diparka340@gmail.com

Under the Guidance of

Abdul Aziz Md

Master Trainer, Edunet Foundation

ACKNOWLEDGEMENT

I would like to take this opportunity to express my deep sense of gratitude to all individuals who helped me directly or indirectly during this thesis work.

Firstly, I would like to thank my supervisor, Abdul Aziz Md, for being a great mentor and the best adviser I could ever have. His advice, encouragement, and constructive criticism have been a source of innovative ideas, inspiration, and the driving force behind the successful completion of this project. The confidence shown in me by him was the biggest source of motivation and inspiration throughout this work. It has been a privilege working with him for the last one year. He always offered assistance not only with the project but also with many other aspects related to the program. His insightful talks and valuable lessons have not only helped in project work but also shaped me into a responsible and competent professional.

ABSTRACT

This project focuses on the development and deployment of a machine learning model for image classification, leveraging both traditional and deep learning techniques. The primary objective is to classify images accurately using models such as MobileNetV2 (for ImageNet classification) and a custom-trained CIFAR-10 model, while comparing their performance in real-world applications. The project aims to contribute to the field of computer vision by showcasing how different image classification models can be applied to varied datasets.

The problem addressed in this project is the need for reliable and efficient image classification systems for real-time applications, where accurate predictions are crucial. Two distinct image classification models are employed: MobileNetV2, a lightweight deep learning model pre-trained on ImageNet, and a custom model trained on the CIFAR-10 dataset, which consists of 10 distinct classes of everyday objects.

The methodology includes image pre-processing techniques such as resizing, normalization, and augmentation. The MobileNetV2 model is used for transfer learning, while the CIFAR-10 model is trained from scratch using TensorFlow. Both models are evaluated using accuracy and confidence scores for their predictions.

Key results include the successful deployment of both models in a user-friendly interface using Streamlit, where users can upload images and receive classification results instantly. The MobileNetV2 model demonstrated strong performance on ImageNet classes, while the CIFAR-10 model showed promising results for classifying objects from the CIFAR-10 dataset.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

Introduction

1.1 Problem Statement:

The problem addressed in this project is the need for efficient image classification in various real-world applications, such as automated object recognition, medical imaging, and security systems. Traditional image classification models often rely on manual feature extraction and are limited in their ability to generalize across diverse image categories. As deep learning models, particularly convolutional neural networks (CNNs), have shown significant promise in tackling image classification tasks, the goal of this project is to leverage pre-trained models like MobileNetV2 and custom models trained on the CIFAR-10 dataset to build a robust, scalable image classification system. The significance of this problem lies in the growing need for accurate and efficient image classification systems in industries ranging from e-commerce to healthcare, where automation of visual data interpretation is critical.

1.2 Motivation:

This project was chosen due to the rapid advancements in deep learning, particularly in the field of computer vision. MobileNetV2, being a lightweight yet powerful model pre-trained on the ImageNet dataset, allows for real-time image classification on devices with limited computational power. The CIFAR-10 dataset provides a well-established benchmark for testing classification models on smaller images with 10 distinct classes. By using these models, the project aims to contribute to the development of automated systems that can handle image classification tasks efficiently, with potential applications in areas like:

E-commerce: Automatically classifying products based on images for sorting and tagging.

Medical Imaging: Assisting doctors by automatically detecting abnormalities in medical scans.

Security: Enhancing object detection capabilities in surveillance systems. The impact of this project lies in its ability to deliver an accessible, practical solution for image classification tasks that can scale across industries.

1.3 Objective:

The primary objectives of this project are:

- **To develop an image classification system that utilizes both pre-trained models (MobileNetV2) and a custom-trained model on the CIFAR-10 dataset.**
- **To evaluate the performance of both models based on key metrics such as accuracy, precision, recall, and F1 score.**
- **To build a user-friendly web application using Streamlit, where users can upload images and get real-time predictions.**
- **To compare the results of MobileNetV2 (ImageNet) and the CIFAR-10 model in terms of prediction accuracy and confidence, highlighting the advantages and trade-offs of each approach.**

1.4 Scope of the Project:

The scope of this project includes:

- **Image Classification Models: The focus is on two models: MobileNetV2, pre-trained on the ImageNet dataset, and a custom model trained on the CIFAR-10 dataset.**
- **Web Application Development: The project will include building a web-based interface using Streamlit, where users can upload an image for classification.**
- **Evaluation and Comparison: The performance of both models will be evaluated using standard metrics, and a comparative analysis will be provided.**

The limitations of the project are:

- **The models are trained on limited datasets (CIFAR-10 and ImageNet), and their performance may vary when applied to images outside the scope of these datasets.**
- **The CIFAR-10 model is restricted to 10 classes, which limits its applicability to more complex tasks requiring a wider range of categories.**
- **The performance of the model may be limited by the computational resources available, particularly for training the CIFAR-10 model or running MobileNetV2 on less powerful devices.**

Literature Survey

SL. NO.	TITLE	OVERVIEW	STRENGTH/ LIMITATION	GAPS	PROJECT ADDRESS
1.	AlexNet: ImageNet Classification with Deep Convolutional Neural Networks	AlexNet demonstrated the power of deep CNNs by winning the ImageNet competition. It highlighted the importance of large neural networks in image classification.	Strength: Achieved significant performance improvements. Limitation: Very large network size, requiring extensive computational resources.	Gap: Computationally expensive, difficult to deploy on edge devices.	We aim to create more efficient models with fewer parameters.
2.	CIFAR-10: A Benchmark for Image Classification Models	CIFAR-10 is a widely-used benchmark dataset for evaluating image classification models. It contains 60,000 32x32 pixel images in 10 different classes.	Strength: Simple and widely used for benchmarking. Limitation: Low resolution of images compared to modern datasets like ImageNet.	Limited to 32x32 images with low resolution.	By using a higher-resolution version and combining it with custom models, we address the need for real-world image classification.

3.	**MobileNetV2: Inverted Residuals and Linear Bottleneck**	**MobileNetV2 is designed for mobile devices, utilizing depthwise separable convolutions to reduce computational cost while maintaining high accuracy.**	**Strength: Lightweight model, suitable for mobile and embedded devices. Limitation: May not achieve the same accuracy as larger models like ResNet.**	**Potential trade-off between accuracy and model size for edge devices.**	**By using MobileNetV2 and fine-tuning it, we ensure both efficiency and performance.**
4.	**ResNet: Deep Residual Learning for Image Recognition**	**ResNet introduced residual connections to combat the vanishing gradient problem, enabling training of much deeper networks.**	**Strength: Solved vanishing gradient problem, allowing deeper networks. Limitation: The architecture may still be computationally expensive.**	**High computational cost for very deep networks.**	**Use of more efficient architectures like MobileNetV2 for deployment on resource-constrained devices.**

Proposed Methodology

3.1 System Design

Here is my proposed System Design :

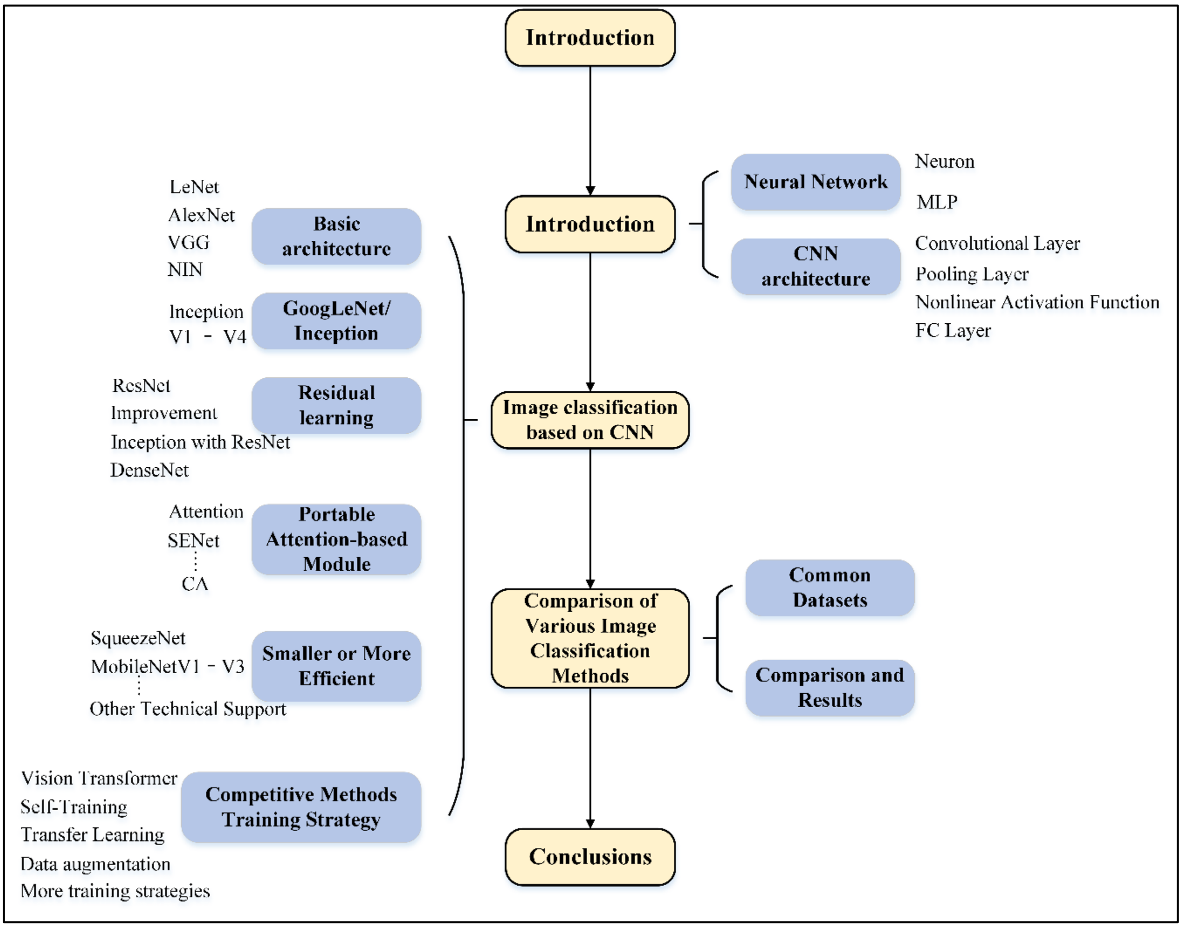


Diagram Explanation:

3.1.1 Introduction Layer: The process starts with introducing the basic concepts of neural networks and CNN architecture.Neural networks are described with components like neurons and MLP (Multi-Layer Perceptron), while CNN architecture is broken down into layers such as convolutional, pooling, nonlinear activation, and fully connected layers.

3.1.2 Basic Architecture Layer: Classic CNN models like LeNet, AlexNet, and VGG are introduced as foundational architectures that have been widely used for image classification tasks.NIN (Network in Network) emphasizes enhancing traditional layer designs with micro multi-layer perceptrons.

3.1.3 Residual Learning Layer: This layer introduces models like ResNet and DenseNet that address challenges like vanishing gradients by utilizing techniques like skip connections and dense connectivity.Variants like Inception with ResNet combine the strengths of multiple architectures.

3.1.4 Attention Mechanism Layer: Attention modules like SENet (Squeeze-and-Excitation) and CA (Coordinate Attention) are described as tools to focus on the most critical regions of the image for improved performance.

3.1.5 Smaller or Efficient Models Layer: Lightweight models such as SqueezeNet and MobileNet are highlighted for their efficiency and suitability for devices with limited resources.

3.1.6 Competitive Methods and Training Strategy Layer: This layer includes modern approaches like Vision Transformers for image classification.Strategies such as self-training, transfer learning, and data augmentation are explained as methods to enhance the model's performance using available data.

3.1.7 Comparison Layer: After building the models, they are compared using common datasets (e.g., ImageNet or CIFAR).Results are analyzed based on performance metrics like accuracy, speed, and model complexity.

3.1.8 Conclusion Layer: The final stage summarizes the advancements and comparisons in CNN-based image classification, highlighting key insights and future trends.

3.2 Requirement Specification:

3.2.1 Hardware Requirements:

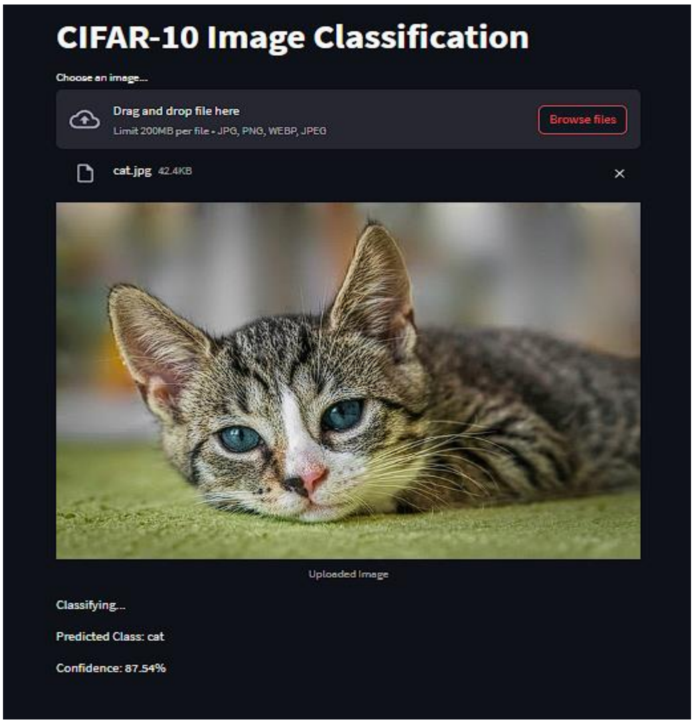
- **CPU: Intel Core i5/i7 or higher (Recommended for general-purpose computation)**
- **RAM: 8GB or more (Recommended to handle image data and model processing)**
- **GPU: NVIDIA GTX 1060 or equivalent (Essential for training deep learning models like MobileNetV2)**
- **Storage: 10GB of free space (For storing models, dataset, and other dependencies)**
- **Input Device: Keyboard, Mouse, and Web Camera (Optional for real-time data input or video-based analysis)**

3.2.2 Software Requirements:

- **Operating System: Windows 10/11 or Linux-based system (e.g., Ubuntu)**
- **Python 3.x: For scripting and running machine learning models**
- **Libraries:**
    - **TensorFlow: For implementing machine learning models (e.g., MobileNetV2, CIFAR-10)**
    - **NumPy: For numerical computations**
    - **Pillow: For image manipulation (resizing, converting formats)**
    - **Streamlit: For building and running the interactive web interface**
    - **Matplotlib: For plotting graphs (e.g., loss curve, accuracy curve)**
    - **Scikit-learn: For additional preprocessing or feature extraction tasks**
- **IDEs/Editors:**
    - **VS Code or PyCharm for coding**
    - **Jupyter Notebook (Optional) for experimentation and testing**
- **Web Browser: Chrome or Firefox for testing the Streamlit interface**
- **CUDA (Optional): For GPU acceleration (for faster training or inference)**
- **Model File (e.g., cifar10_model.h5): Pre-trained model file for classification**

Implementation and Result

4.1 Snap Shots of Result:



4.2 GitHub Link for Code:

https://github.com/Creating-Content/Image-Classification-using-ML-and-DL

Discussion and Conclusion

5.1 Future Work:

While the model used for image classification in this project demonstrates strong accuracy in recognizing and classifying images, there are several areas for future enhancement:

I. **Improvement in Accuracy: Future work could focus on using more advanced models like EfficientNet or Vision Transformers (ViT), which might offer better performance and efficiency compared to MobileNetV2 or simpler CNN models.**
II. **Data Augmentation: Incorporating more diverse datasets and leveraging advanced data augmentation techniques could improve the model's ability to generalize to unseen images, particularly for specific classes.**
III. **Transfer Learning: While the project leverages a pre-trained MobileNetV2 model, fine-tuning it on more domain-specific datasets (like CIFAR-10 or specialized medical images) can potentially boost performance.**
IV. **Real-Time Classification: Further optimizing the model for real-time image classification by improving inference speed and reducing model size (perhaps through pruning or quantization) can make it more suitable for embedded or edge devices.**
V. **Explainability: Introducing methods like Grad-CAM or SHAP to improve model interpretability, making it easier to understand how predictions are made, especially for critical applications such as medical diagnostics.**
VI. **Integration with Other Modalities: Combining image classification with other modalities like text or sensor data (multimodal systems) could be explored to increase the robustness and versatility of the system.**

5.2 Conclusion:

This project presents a machine learning and deep learning-based solution for image classification, specifically utilizing pre-trained models like MobileNetV2 and CIFAR-10 to classify images accurately. The system demonstrates the power of modern CNN architectures and transfer learning in achieving high-performance classification with relatively small datasets. Despite the promising results, the project also highlights the need for further refinement in terms of real-time performance, model interpretability, and generalization across diverse image categories. By focusing on improving these aspects, future work will contribute to building more robust, efficient, and transparent image classification systems with broader real-world applications. The ultimate impact of this project lies in its potential to assist in automating and enhancing various tasks that require visual recognition, offering benefits across industries such as healthcare, security, and e-commerce.

REFERENCES

[1] Ming-Hsuan Yang, David J. Kriegman, Narendra Ahuja, "Detecting Faces in Images: A Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 34-58, 2002.

[2] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 25, pp. 1097-1105, 2012.

[3] Karen Simonyan and Andrew Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015.

[4] Yann LeCun, Yoshua Bengio, Geoffrey Hinton, "Deep Learning," *Nature*, vol. 521, pp. 436-444, May 2015.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Deep Residual Learning for Image Recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770-778, 2016.

[6] Olaf Ronneberger, Philipp Fischer, Thomas Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 234-241, 2015.