SANTA CLARA UNIVERSITY

Machine Learning

COEN 240 – FALL 2023

PROJECT REPORT

DHEERAJ SAI VENKAT GEDUPUDI

Abstract:

This project develops a robust face recognition system utilizing a pre-trained ResNet50 Convolutional Neural Network. Aimed at processing diverse image formats (HEIC, PNG, JPEG), the system standardizes images to a uniform size for effective feature extraction and classification. The use of ResNet50 facilitates accurate identification and categorization of faces, highlighting the system's potential for practical applications in security and digital media.

Introduction:

Background

Face recognition technology has become a cornerstone in various applications, ranging from security to digital media. The advent of deep learning, particularly Convolutional Neural Networks (CNNs), has significantly advanced the capabilities in this domain.

Project Overview

This project explores the application of CNNs for face recognition, with a specific focus on employing a pre-trained ResNet50 model. The ResNet50 model is renowned for its deep learning efficiency and accuracy in image recognition tasks.

Objective and Challenge

The primary objective is to create a system capable of accurately recognizing and categorizing faces from images in varied formats. A significant challenge is the standardization of these diverse image formats for consistent processing. By converting all images to a standardized 224 x 224 JPEG format, the project ensures uniformity essential for the subsequent application of the ResNet50 model.

Significance

The project's significance lies in leveraging the advanced capabilities of a pre-trained CNN, minimizing the need for extensive training data and resources while maintaining high accuracy in face recognition. This approach is particularly valuable in practical scenarios where reliable and efficient face recognition is essential.

Project Workflow Overview:

Data Collection and Preprocessing

- Source of Data: The dataset, named “Release,” was initially provided by Professor Chen.
- Data Cleaning and Standardization: Given the variety in file formats (JPEG, PNG, HEIC, etc.) and discrepancies in aspect ratios and resolutions, all images were converted to a uniform size of 224 x 224 pixels in JPEG format.

Face Detection and Image Preprocessing

- Face Detection: Utilized OpenCV’s Haar Cascade Classifier for detecting faces within the images.
- Image Cropping and Resizing: The detected faces were isolated, cropped, and resized into 224 x 224 JPEG files.

Dataset Splitting

- Training and Validation Split: An 80/20 split was implemented, with 80% of the dataset allocated for training and the remaining 20% for validation.

Dataset Augmentation

- Flipping the Images: To enhance the dataset, each image was flipped horizontally, effectively doubling the size of the dataset.

Model Selection and Training

- Choice of Model: The pre-trained ResNet50, a 50-layer deep CNN, was chosen due to its proven effectiveness in image classification tasks and its ability to yield high-quality results.
- Model Training: Training was conducted over 10 epochs using the Adam Optimizer, which is known for its adaptive learning rate capabilities. This optimizer excels in handling sparse gradients and ensures quick convergence.

Model Evaluation

- Validation Process: The model was validated using the set aside validation dataset.
- Performance Metrics: The model achieved 100% accuracy on the training dataset, 95.41% on the validation dataset, and 89.84% on the testing dataset.

Conclusion

The project successfully implemented a face recognition system using a pre-trained ResNet50 model, achieving impressive accuracy rates across training, validation, and testing datasets. The effectiveness of the CNN in handling diverse image formats and the robustness of the preprocessing and augmentation strategies were key to these results.

Challenges Faced and Solutions:

Inconsistent Face Detection

Challenge: The Haar Cascade Classifier initially failed to detect faces in some images.
Solution: Images where faces were undetected were set aside for future refinement. This approach allowed for focusing on images where face detection was successful, ensuring the project's progression.

Multiple Face Detections

Challenge: The classifier often detected multiple faces in a single image.
Solution: To resolve this, a logic was implemented to select only the largest detected face. This approach assumed that the largest bounding box was most likely to be the relevant face, thus simplifying the dataset for more effective training.

Misidentification of Non-Facial Features

Challenge: The classifier occasionally misidentified non-facial features (like designs on shirts) as faces.
Solution: To improve accuracy, images were manually reviewed and cropped to ensure only faces were included. While labor-intensive, this step significantly increased the training dataset's quality.

Inconsistency in Dataset Labels

Challenge: A discrepancy was observed in the labeling convention between the initially released dataset and the later released testing dataset, particularly concerning white spaces in names.
Solution: To standardize the dataset, all white spaces between first names and last names were removed during preprocessing. This adjustment ensured consistency across the datasets, allowing for more accurate labeling and classification.

Preprocessing Stage:

Initial Data Processing

- File Format Handling: The dataset comprised various file formats, including PNG, HEIC, JPEG, and JPG. The pyheif library was employed to efficiently read HEIC files, ensuring all images, regardless of format, were accessible for processing.
- Face Detection: OpenCV's Haar Cascade Classifier was utilized to detect faces within the images. To ensure the selection of the most relevant face, only the largest detected bounding box was chosen. This strategy helped in focusing on the primary subject in each image.

Image Cropping and Resizing

- Cropping Process: After detecting the faces, the images were cropped to include only the facial area. This step was crucial in eliminating irrelevant background information and standardizing the focus on faces.
- Standardization: Cropped images were then resized and saved in a new directory as 224 x 224 JPEG files. This uniform size and format were essential for consistent model input.

Data Cleaning and Quality Control

- Handling Discrepancies: Several discrepancies, such as misidentified non-facial features and inconsistent labeling, were identified and rectified. The details of these challenges and their solutions are extensively discussed in the 'Challenges Faced and Solutions' section of this report.

Dataset Splitting and Augmentation

- Splitting the Dataset: The dataset was divided into training and validation sets with an 80/20 split. This separation ensured a substantial amount of data for model training while reserving a portion for model validation.
- Augmentation Strategy: To enhance the dataset and prevent overfitting, each image in the dataset was flipped horizontally, effectively doubling its size. Importantly, this augmentation was performed after splitting the dataset into training and validation sets, as advised by the professor. This approach prevented skewed distribution of flipped and unflipped images in the training and validation datasets, ensuring a more robust and generalizable model.

Feature Extraction and Model Training:

ResNet50 for Feature Extraction

- Pre-Trained Model: The ResNet50 model, pre-trained on the ImageNet dataset, was utilized for its robust feature extraction capabilities. This pre-training provided a solid foundation of learned image features, which is beneficial for the face recognition task.
- Modification of Architecture: The top layer of the ResNet50 was excluded to adapt the model to the specific requirements of this project. In its place, a Global Average Pooling layer was added to reduce overfitting by minimizing the total number of parameters in the model.
- Task-Specific Adaptation: Following the pooling layer, custom Dense layers were introduced. These layers are crucial for tailoring the model to the specific task of face recognition.

Custom Dense Layers

- Design of Layers: The custom Dense layers use ReLU (Rectified Linear Unit) activation, known for its effectiveness in non-linear transformations within neural networks. This choice helps in capturing complex patterns in the data.
- Final Output Layer: The model concludes with a softmax output layer, which is standard for multi-class classification problems. This layer transforms the output into probability scores for each class, facilitating easier interpretation of the model's predictions.

Adam Optimizer

- Choice of Optimizer: The Adam Optimizer was chosen for its adaptive learning rate properties. This optimizer adjusts the learning rate throughout training, which can lead to faster convergence and better overall performance.
- Handling Sparse Gradients: Adam is particularly effective in dealing with sparse gradients, a common challenge in deep learning models, ensuring that the model remains efficient during training.

Future Steps:

Improved Face Detection:
Future iterations of the project will focus on implementing a more advanced face detection algorithm. An algorithm that would ideally assign probabilities to each detected face, allowing for the selection of the most probable face instead of merely the largest one, thereby enhancing the accuracy of the system.

Further Dataset Expansion:
Exploring additional data augmentation techniques or incorporating more varied datasets could further improve the model's robustness and generalization capability.

Real-time Application Development:
Developing a real-time face recognition application, possibly integrating the system into security or personalized user experience platforms.
This future direction aims to refine the system’s accuracy and broaden its practical applicability, ultimately leading to a more sophisticated and versatile face recognition tool.