Introduction

In this project, we aim to predict the age of individuals from their images. We utilize a deep learning model based on the ResNet50 architecture to perform this regression task. The dataset comprises images and their corresponding age labels.

Libraries and Dependencies

We use several libraries including pandas, seaborn, matplotlib, and TensorFlow. TensorFlow is used for building and training the deep learning model, while pandas is used for data manipulation, and seaborn and matplotlib are used for data visualization.

Data Exploration and Pre-processing

1. Loading Data

- The dataset is loaded from a CSV file named '**age_gender.csv**' using pandas.
- A quick examination of the data is done using the '**head()**' method to understand its structure.

2. Visualizing Age Distribution

- A distribution plot of ages is created to understand the age distribution in the dataset.

3. Image Pre-processing

- The pixel values of images are extracted, reshaped, and saved as individual image files in a directory.
- TensorFlow's '**ImageDataGenerator**' is used to create a data generator for training the model. This generator is configured to rescale the pixel values of the images.

4. Displaying Images

- A batch of images is retrieved from the data generator and displayed along with their true age labels to verify the data preparation steps.

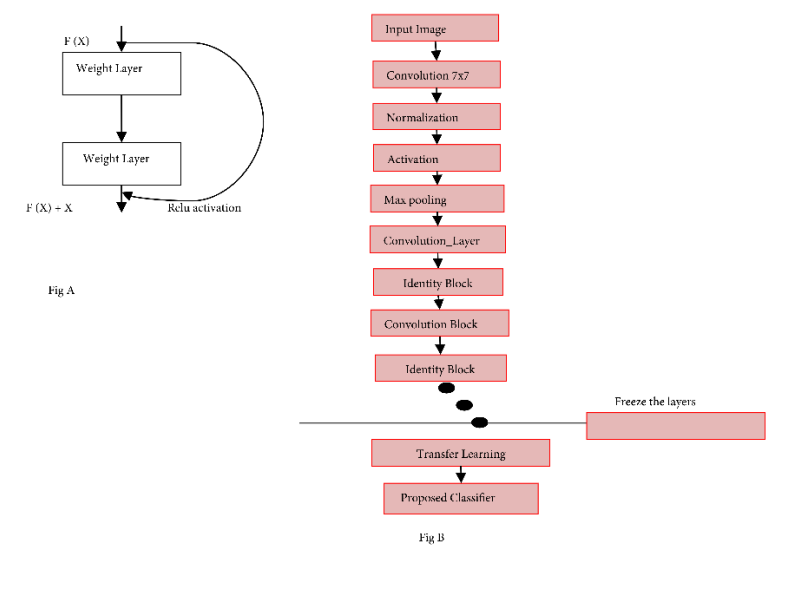
Model Development

1. Data Preparation

- Two functions '**load_train**' and '**load_test**' are defined to load the training and validation datasets respectively using '**ImageDataGenerator**'.

2. Model Architecture

- The model is based on the ResNet50 architecture with the top layers removed.
- Custom layers including Dropout, GlobalAveragePooling2D, and a Dense layer are added to the model.
- The model is compiled using the Adam optimizer and Mean Absolute Error (MAE) as the loss function since this is a regression task.



Model Architecture

3. Training

- The '**train_model**' function is defined to train the model using the training data and validate it using the validation data.
- The model is trained for a specified number of epochs, and the training history is returned for further analysis.

Evaluation

1. Sample Prediction

- A sample image is selected from the training data, and a prediction is made using the trained model. The predicted and true ages are displayed along with the image.

2. Model Evaluation

- The model is evaluated on the validation data to obtain the Mean Absolute Error (MAE) on the validation dataset.
- Predictions are made on the validation data, and an accuracy percentage is calculated based on a tolerance value.

3. Visualization

- The training and validation loss and MAE are plotted over epochs to visualize the training process and understand the model's performance.

Hyperparameter Tuning

1. Setup

- The Keras Tuner library is used for hyperparameter tuning.
- A custom hypermodel class is defined which builds a model based on given hyperparameters.

2. Tuning

- A Random Search is performed over specified ranges of dropout rates and learning rates to find the optimal hyperparameters for the model.
- The search is performed for a specified number of trials, and the best hyperparameters are displayed at the end.

Conclusion

This project demonstrates the process of building, training, and evaluating a deep learning model for age prediction from images. The steps include data exploration, pre-processing, model development, training, evaluation, and hyperparameter tuning. Through this project, we gain insights into the age distribution in the dataset, the performance of the model, and the optimal hyperparameters for training the model.