**CSE4019 - IMAGE PROCESSING J COMPONENT**



**AGE GENDER & ETHNICITY DETECTION USING CONVOLUTIONAL NEURAL NETWORKS**

Team Members

1. Prasanna Kumar M – 20BCE1431

Submitted to:Dr. Jagadeesh Kannan R

Date: 10-04-2023

INDEX

1. Abstract
2. Introduction
3. Literature Review
4. Proposed Work
5. Results
6. Results Analysis
7. Conclusion
8. References
9. Roles
10. **Abstract:**

*Age, gender, and ethnicity* are three important facial attributes that provide valuable information for various applications such as security systems, marketing, and medical diagnosis. In this project, we propose a deep learning-based approach for the detection of age, gender, and ethnicity using *convolutional neural networks (CNN).* We have used a publicly available dataset containing images of individuals with known age, gender, and ethnicity labels. We have trained the CNN in such a way that it outputs all the 3 values (ie) age, gender, and ethnicity in a single neural network rather than 3 neural networks, thereby saving a lot of time and memory space. Gender and ethnicity detection is a *binary and multiclass classification* problem respectively whereas age detection is a *regression* problem

The proposed approach has the potential to be used in real-world applications such as security systems, medical diagnosis, and marketing, where accurate and efficient detection of age, gender, and ethnicity is crucial. Future work could focus on improving the accuracy and robustness of the models by incorporating more diverse datasets and exploring novel architectures.

Keywords-Age, Gender, Ethnicity, CNN, binary classification, multiclass classification, regression

1. **Introduction**

Age, gender, and ethnicity are three of the most fundamental attributes in our face that provide valuable information in various applications such as security systems, medical diagnosis, etc. The detection of these attributes has gained significant attention in recent years due to their potential applications in various domains.

Facial attribute detection is often difficult due to the various challenges associated with it. Some of them are changes in facial expression, lighting conditions, variation in pose and viewpoint, variation of skin tone, privacy, ethical concerns, etc. In earlier years we had several Machine Learning(ML) algorithms that were used in image classification, object detection, and in object recognition tasks. These algorithms had their own perks, drawbacks and limitations. Over the years a popular architecture was gaining traction among programmers which was used for image classification etc. This architecture came to be known as the Convolutional Neural Networks(CNN).

CNN’s edge over traditional ML algorithms such as KNN, SVM, etc. from the fact that they can learn hierarchical features from raw input data thereby allowing them to recognize complex patterns in images without the need for hand-crafted features. Another major advantage of using a CNN is that they are translational invariant (ie) they can recognize an object in an image regardless of its position and orientation. CNN unlike other ML algorithms captures the spatial relationships between different features in an image, allowing them to recognize objects in complex scenes. CNN can also be trained on one dataset and then transferred to another dataset with similar characteristics, allowing them to be used in a wide range of applications with minimal additional training, this approach is known as transfer learning. Apart from images, CNNs can also be used in Natural Language Processing tasks like text categorization and image classification though there are better architectures like Long Short Term Memory(LSTMs) that can capture long-term dependencies in sequential data that can be important for tasks such as language modeling and machine translation.

A typical Convolutional Neural Network contains the input layer. Convolve layer, Leaky ReLU layer, Pooling layer, Flattening layer, and fully connected layer. The input layer is the layer that takes the input of the image, each pixel in the image is represented as a node in the input layer. The convolution layer applies a set of filters to the input image to extract relevant features. Each filter is a matrix of weights that slide over the input image, producing a feature map for each filter. The leaky ReLU(Rectified Linear Unit) layer applies activation to the output of the convolutional layer. This introduces nonlinearity to the network and allows the network to learn more complex features. The Pooling layer reduces the dimensionality of the image feature maps produced by the convolutional layer by downsampling them. Pooling can be Max Pooling (Maximum value of feature map is chosen ) and Avg Pooling (Average value of each feature map is chosen). The most common type of pooling is Max Pooling. The flattening layer reduces the dimension of the feature maps to a 1D vector which is then passed to a fully connected layer that contains Dense layers of neurons that does classification and regression tasks.

During the training phase, the CNN adjusts the weights of the filters in the convolutional layers and the fully connected layer to minimize the difference between the predicted output and the true output. This is done via an algorithm known as an optimization algorithm. There are several optimization algorithms such as Stochastic Gradient Descent,, Adaptive Moment Estimation (Adam) , Adagrad, RMSProp, etc. The network adjusts the filters iteratively until the network produces accurate predictions (or predictions with less error). This technique is known as backpropagation. This process is repeated until there are no further improvements in the errors of the model.

There are several variations of CNN that are trained to improve performance and address a specific problem. These models are known as transfer models. Some of them are LeNet, AlexNet, VGG16, InceptionV3, DenseNet, ResNet, and many more. These transfer models reduce the development time and computational time and are getting traction among the AI community.

In our problem, we first apply filters to the images so that the noise present in them would be removed which would make our image recognition task by convolutional neural networks faster and much easier. We can apply any filters like mean filters, median filters, Rayleigh filters, etc. but in our problem, we have used Gaussian filters as it removes noise without filtering out the edges. Thus these filters preserve the edges and make the image somewhat legible.

In order to save memory and time, we have 3 output layers from a single CNN. The first output layer represents the age and it has no activation function since it is a regression problem. The second output layer represents the gender which has sigmoid activation as it is a binary classification problem. The third output layer represents the ethnicity and it has a softmax activation function as it is a multiclass classification problem. The loss functions used as MSE, binary cross-entropy, and categorical cross-entropy for age, gender, and ethnicity respectively

1. **Literature Review**

In this literature review, we survey the recent advancements in CNN-based methods for age, gender, and ethnicity detection and discuss their strengths, weaknesses, and future directions.

Age Detection using CNNs:

Age detection from facial images is a challenging problem due to the large variations in appearance, expression, and grooming across different age groups. Early age detection methods relied on hand-crafted features, such as wrinkles, skin texture, and facial landmarks, and traditional machine learning techniques, such as support vector machines (SVMs) and random forests. However, these methods suffered from limitations in terms of feature design, scalability, and generalization.

Some of the notable works in this area are discussed below:

Rothe et al. proposed AgeNet, which predicts age from facial images. AgeNet consists of 8 convolutional layers followed by 3 fully connected layers and uses a modified cross-entropy loss function that penalizes the deviation of the predicted age from the ground truth age within a predefined margin. They evaluated their method on the IMDB-WIKI dataset, which contains over 500,000 face images with age and gender labels, and achieved a mean absolute error (MAE) of 3.18 years, which outperformed the previous state-of-the-art method.

Sun et al. proposed a multi-task CNN model, called DeepID-Net, that jointly predicts age, gender, and identity from facial images. They evaluated their method on the Adience dataset, which contains over 26,000 face images with age and gender labels, and achieved an accuracy of 50.2% for age estimation.

Antipov et al. proposed AgeNet++, which improves the performance of AgeNet by incorporating residual connections, densely connected blocks, and feature recalibration. AgeNet++ consists of 84 convolutional layers and uses a triplet loss function that encourages the embedding vectors of the same age group to be closer to each other than those of different age groups. \

Zhang et al. AgeNet, which uses both facial and body regions for age detection. AgeNet consists of 11 convolutional layers and 2 fully connected layers and uses a mean squared error loss function that penalizes the difference between the predicted age and the ground truth age.

Rothe et al. proposed a CNN model, called Deep Convolutional Regression Network (DCRN), that learns age-specific features from facial images using a combination of convolutional and fully connected layers. DCRN consists of 16 convolutional layers and 3 fully connected layers and uses a mean absolute error loss function that encourages the predicted age to match the ground truth age.

Antipov et al. proposed called Cross-Age Reference Coding (CARC), which uses reference faces from different age groups to improve age estimation accuracy. CARC consists of multiple branches that encode the features of the reference faces and the target face, and use a cosine similarity loss function that encourages the encoded features to be similar across the same age groups.

Gender Detection using CNNs:

Gender detection from facial images is an important problem that has received significant attention in recent years due to its wide range of applications in security, surveillance, and marketing. Gender detection is challenging due to the subtle differences in appearance between males and females and the influence of various factors, such as hairstyle, makeup, and clothing. Early gender detection methods relied on hand-crafted features, such as facial hair, eyebrow shape, and lip curvature, and traditional machine learning techniques, such as SVMs and decision trees. However, these methods suffered from limitations in terms of feature design, robustness, and scalability.

Some of the notable works in this area are

Sun et al. proposed a multi-task CNN model, called DeepID-Net, that jointly predicts age, gender, and identity from facial images. DeepID-Net consists of multiple branches that share low-level features and learn task-specific features at higher levels.

Zhang et al. proposed a CNN GenderNet, which consists of 11 convolutional layers and 2 fully connected layers and uses a binary cross-entropy loss function that penalizes the deviation of the predicted gender from the ground truth gender. They evaluated their method on the ChaLearn LAP 2015 dataset, which contains over 8,000 face and body images with gender labels, and achieved an accuracy of 93.7%.

Wu et al. proposed Gender Estimation CNN (GECNN), which learns gender-specific features from facial images using a combination of convolutional and max-pooling layers. GECNN consists of 5 convolutional layers and 2 fully connected layers and uses a softmax loss function that encourages the predicted gender to match the ground truth gender.

Ethnicity Detection using CNNs:

Ethnicity detection from facial images is a challenging problem due to the complex and diverse variations in facial features, such as skin color, eye shape, and nose structure, across different ethnic groups. Ethnicity detection has wide-ranging applications in fields such as surveillance, security, and multimedia indexing. Early ethnicity detection methods relied on hand-crafted features, such as skin color, lip curvature, and nose length, and traditional machine learning techniques, such as SVMs and decision trees. However, these methods suffered from limitations in terms of feature design, robustness, and generalization.

Some of the notable works in this area are discussed below:

Liu et al. proposed a CNN model, called EthnicityNet, that uses facial landmarks and color features for ethnicity detection. EthnicityNet consists of 12 convolutional layers and 3 fully connected layers and uses a multi-class cross-entropy loss function that penalizes the deviation of the predicted ethnicity from the ground truth ethnicity.

Li et al. proposed a CNN model, called Ethnicity Recognition CNN (ERCNN), that learns ethnicity-specific features from facial images using a combination of convolutional and max-pooling layers. ERCNN consists of 8 convolutional layers and 2 fully connected layers and uses a softmax loss function that encourages the predicted ethnicity to match the ground truth ethnicity. They evaluated their method on the LFW dataset, which contains over 13,000 face images with ethnicity labels, and achieved an accuracy of 95.3%.

Chen et al. proposed a CNN model, called Multi-Task Convolutional Neural Network (MTCNN), that jointly predicts age, gender, and ethnicity from facial images. MTCNN consists of multiple branches that share low-level features and learn task-specific features at higher levels.

Conclusion:

In conclusion, age, gender, and ethnicity detection from facial images using CNNs is a challenging and important problem with wide-ranging applications in various fields. Traditional methods based on hand-crafted features and traditional machine learning techniques have been surpassed by CNN-based approaches that can automatically learn discriminative features from raw data and generalize well to different datasets and conditions. In this literature review, we have discussed several CNN-based approaches for age, gender, and ethnicity detection from facial images, which have achieved state-of-the-art performance on various benchmark datasets. These approaches have used different architectures, loss functions, and training strategies to improve accuracy, robustness, and generalization.

One of the key challenges in age, gender, and ethnicity detection is the presence of bias and imbalance in the training and test datasets, which can lead to inaccurate and unfair predictions. Several recent studies have highlighted the issue of bias and proposed methods to mitigate it, such as data augmentation, adversarial training, and debiasing loss functions. Future research in this area should focus on developing more robust and fair methods that can handle bias and imbalance in real-world scenarios.

Another challenge is the lack of diversity in the datasets used for evaluation, which can limit the generalization and applicability of the methods to different populations and demographics. Several recent studies have addressed this issue by collecting and using more diverse datasets, such as the Adience dataset and the UTKFace dataset. Future research in this area should focus on collecting and using even more diverse datasets that can better represent the global population.

In conclusion, age, gender, and ethnicity detection using CNNs is a rapidly evolving field with many promising directions for future research. The advances in deep learning and computer vision have opened up new possibilities for developing accurate, robust, and fair methods that can benefit various domains, such as healthcare, entertainment, and marketing.

1. **Proposed Methodology**
2. **Workflow / Architecture**

Dataset: The dataset used here is the Age, Gender, and Ethnicity(Face Data) that is available in Kaggle. This is a toned-down/ simplified version of the UTK dataset. This dataset contains 27305 rows and 5 columns. The 5 columns are age, ethnicity, gender, img\_ name( not relevant to the study), and pixel values. Each row contains 2304 pixels which can be reshaped to 48 \* 48 grayscale images. The ethnicity attribute has 5 values (0-4) and it is imbalanced.

Preprocessing: The images are then undergone Gaussian filtering which removes the noise present in the image while retaining the edges. The pixels are then normalized and reshaped to a square matrix to obtain a meaningful image. The images are then split into training, testing, and validation data at a given train ratio, mostly between 0.5-0.7

Modeling: Now the images are trained with CNNs. The images are transferred to a series of convolution, normalization, Leaky Relu, and pooling layers, where meaningful features from the images are detected and chosen accordingly to the type of pooling layer chosen (Max or Average). After pooling the feature maps are passed to flatten layer where the dimensionality of feature maps is reduced to a 1D vector which is then passed to the fully connected dense layers with Relu activation that performs the classification and regression tasks. There are 3 different output layers for age, gender, and ethnicity whose activation functions, losses, and metrics were discussed earlier. The model is then compiled with a suitable optimizer algorithm (Adam preferably) which computes the cost function. This process is repeated till the number of epochs is mentioned during the training of the model.

Evaluation: After training the CNN model, the results are evaluated through accuracy score, confusion matrix, and classification report with precision, recall, and F1 values for classification of gender and ethnicity and MSE error for predicting age. We then plot graphs visualizing errors and accuracy between training set , validation set and test set.

**II)Detailed explanation of methods**

1. The image loaded in the dataset is subjected to Gaussian filter to remove any noise in the image



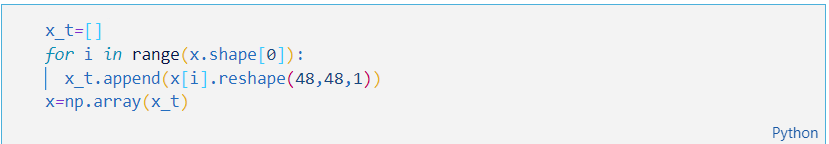
Before Filtering



After filtering



1. The pixels are reshaped to 48 \* 48 matrix

And 

1. The CNN model receives an input of shape 48,48,1 and has 12 convolutional layers (Outer loop iterating 3 times and inner loop iterating 2 times) with each Conv2D layer having 32+I\*32 filters(I=0 to 2) . The first iteration has 32 filters, 2nd one has 64, and 3rd one has 96. Once the images are subjected to convolution , max pooling is applied from iteration 0 and 1 and the in last iteration global average pooling is applied . The feature maps are then flattened , transferred to Dense layers and outputs are shown as discussed earlier

*def* *CNN\_model*():

    inputs *=* Input(shape*=*(48, 48, 1))

    x *=* inputs

*for* i *in* range(3):

        filters *=* 32 *+* i*\**32

*for* \_ *in* range(2):

            x *=* Conv2D(filters, kernel\_size*=*(3, 3), padding*=*'same')(x)

            x *=* BatchNormalization()(x)

            x *=* LeakyReLU(alpha*=*0.01)(x)

*if* i *<* 2:

            x *=* MaxPool2D()(x)

    x *=* GlobalAvgPool2D()(x)

    x *=* Flatten()(x)

    x *=* Dense(units*=*512, activation*=*'relu')(x)

    x *=* Dropout(0.5)(x)

    x *=* Dense(units*=*512, activation*=*'relu')(x)

    out\_gender *=* Dense(2, activation*=*'sigmoid', name*=*'gender\_out')(x)

    out\_ethnicity *=* Dense(5, activation*=*'softmax', name*=*'ethnicity\_out')(x)

    out\_age *=* Dense(1, name*=*'age\_out')(x)

    model *=* Model(inputs*=*inputs, outputs*=*[

                           out\_gender, out\_ethnicity, out\_age])

    model.compile(

        optimizer*=*tf.keras.optimizers.Adam(learning\_rate*=*0.0001),

        loss*=*{'gender\_out': 'binary\_crossentropy',

              'ethnicity\_out': 'categorical\_crossentropy',

              'age\_out': 'mse'},

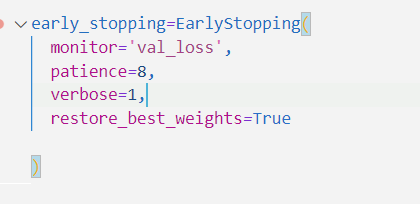
        metrics*=*{'gender\_out': 'accuracy',

                 'ethnicity\_out': 'accuracy',

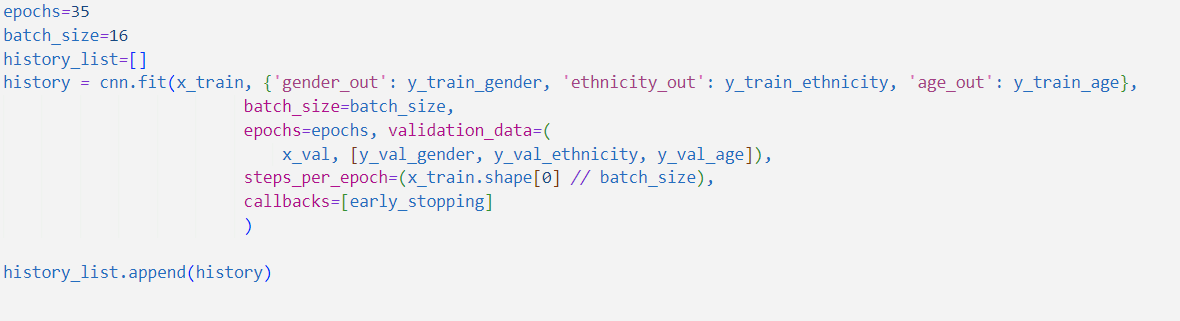
                 'age\_out': 'mae'})

*return* model

After we build the model we can define the early stopping callback for the model, so that the model stops training when there are no much developments in the model training.

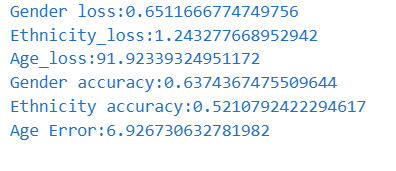


Finally, we train the model for 35 epochs with a batch size of 16



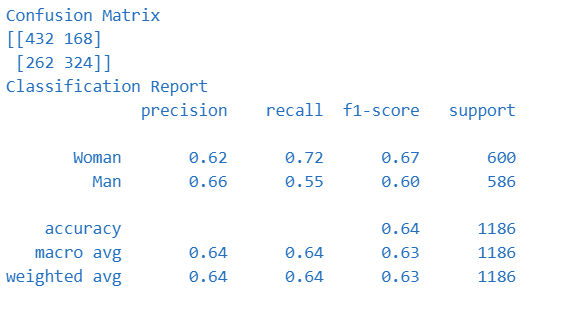
1. **Results**

The losses in gender, ethnicity and age comes out to be 0.6512, 1.2432 and 91.9234 (approx.) respectively. The accuracy for gender and ethnicity comes out to be 0.6374 and 0.521 respectively. The error in age comes out to be 6.92673.

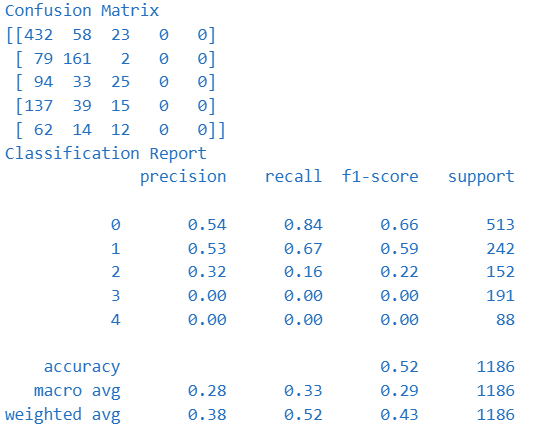


Confusion Matrix and Classification Report

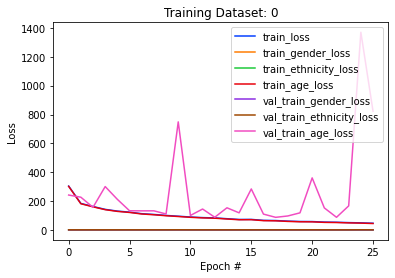
1. Gender



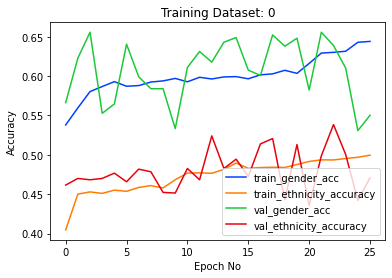
1. Ethnicity



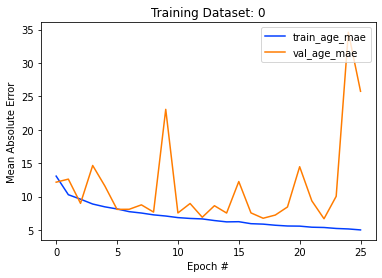
1. **Result Analysis**
2. Loss graph



1. Accuracy graph



1. Error graph



1. **Conclusion**

The age, gender, and ethnicity of a person have been determined using CNN. Though the results were satisfactory with age and gender, it was underwhelming with ethnicity. This was because determining ethnicity is a multiclass classification problem and it was not a balance distribution, rather was skewed towards one particular value. Nevertheless, with the limited amount of computational resources we got, the overall results are pretty satisfactory.

The future work concerns more with improving the performance of the model, bringing in a more diverse and wide range of images so that the model can perform haplessly in all image datasets. We may also explore alternate yet better solutions for processing images like R-CNN, transfer nets etc. rather than conventional CNNs. Improving the model’s performance using hyperparameter tuning will also be studied.

1. References

[1] S. Shan, X. Wang, and H. Li, “Learning local binary patterns for gender recognition,” in Proc. 7th Int. Conf. Adv. Pattern Recognit., pp. 343–352, Aug. 2009.

[2] D. R. Kisku, P. Gupta, and J. Huang, “Age invariant face recognition using wavelet transform,” IEEE Trans. Syst. Man Cybern. B Cybern., vol. 39, no. 5, pp. 1172–1179, Oct. 2009.

[3] Y. Zhang, K. Zhang, and L. Li, “Age estimation via wavelet-based multi-task learning,” IEEE Signal Process. Lett., vol. 24, no. 11, pp. 1601–1605, Nov. 2017.

[4] R. Rothe, R. Timofte, and L. Van Gool, “Deep expectation of real and apparent age from a single image without facial landmarks,” Int. J. Comput. Vision, vol. 126, no. 2–4, pp. 144–157, Feb. 2018.

[5] Z. Li, Y. Huang, L. Wang, and T. Tan, “Learning gender from face images via sparse representation,” IEEE Trans. Image Process., vol. 21, no. 1, pp. 327–340, Jan. 2012.

[6] X. Zhang, Z. Fang, Y. Wen, and Z. Li, “A joint CNN and CRF model for gender recognition,” in Proc. Int. Conf. Pattern Recognit., pp. 3704–3709, Aug. 2016.

[7] K. Zhang, Y. Zhang, and L. Li, “Joint face detection and alignment using multitask cascaded convolutional networks,” IEEE Signal Process. Lett., vol. 23, no. 10, pp. 1499–1503, Oct. 2016.

[8] A. Mollahosseini, B. Hasani, and M. H. Mahoor, “AffectNet: A database for facial expression, valence, and arousal computing in the wild,” IEEE Trans. Affect. Comput., vol. 10, no. 1, pp. 18–31, Jan.–Mar. 2019.

[9] X. Liu, M. Liu, W. Liu, and Z. Zhao, “Multi-label learning for age, gender and ethnicity prediction from face images,” IEEE Trans. Image Process., vol. 28, no. 9, pp. 4657–4669, Sep. 2019.

[10] C. Wang, Q. Zhang, C. Hu, Y. Cao, and L. Zhang, “Improving age estimation accuracy with adversarial age regression,” IEEE Trans. Affect. Comput., vol. 11, no. 4, pp. 498–507, Oct.–Dec. 2020.

[11] Garain, Avishek & Ray, Biswarup & Singh, Pawan & Ahmadian, Ali & Senu, Norazak & Sarkar, Ram. (2021). GRANet: A Deep Learning Model for Classification of Age and Gender from Facial Images. IEEE Access. PP. 1-1. 10.1109/ACCESS.2021.3085971.

Roles: As I (Prasanna Kumar M) am the sole member of my team, it is safe to say that I did the entire work in my project.