Table of Contents

Gender & Ethnicity Detection System Using CNN

Vijay K. Kshetri (01890086), Yerandy Minaya (01318967), Pardeep Pathania (01883876)

Dr. Daniel Shao
Instructor for CIS 602

Department of Computer and Information Science
University of Massachusetts – Dartmouth

Abstract

The research and technology industry are constantly looking to effectively use models to identify certain features that help automate the process of identifying a person or object. Many companies like Facebook, Amazon, and Google are a step ahead and providing this type of solution, not only in the sense of security systems but for other types of business purposes. With the increase of technological developments and curiosity for computer vision, there are many ways to create models such as to detect the biological gender or ethnicity of an individual. However, there has been some backlash for the use of this technology due to gender and ethnicity biases used by law enforcement systems and vendors. These companies are working full time on improving their algorithm to conquer these biases, but identifying a person's gender or ethnicity is a normal process that most people carry out without thinking about it. On the other hand, it is not always easy for people to be able to identify someone's gender or ethnicity solely based on appearances. So, imagine how challenging this could be for a machine. This project aims at evaluating the application of convolutional neural networks (CNN) towards automated detection of the biological gender and ethnicity of a person based on the person's face. Using the UTKFace dataset, which has many types of images to cover the most common semantic gaps, the trained model shows that it can accurately identify the gender and ethnicity of any individual.

Introduction

The rapid progress of AI (Artificial Intelligence) development in Computer Science has led us to the stage where we are now capable of doing the tasks seamlessly that were once extremely difficult or impossible just a few decades ago. Whether it is Cortana or automation to vehicles or robotics, this progress of AI development is not showing any sign of decreased evolution and development. With the use of AI, one can encompass anything with the help of robotics or advanced algorithms. Using the main two approaches of AI, Deep Learning and Machine Learning has made the jobs

even more natural to perform. The vast improvements in these areas have led to a paradigm shift in all the sectors of the industry such as Information Technology, Healthcare, Banking, and many more. From image recognition to video analytics, human-level performance has been approached in visual perception tasks using Computer Vision algorithms.

For machines, mimicking the human eye is not as easy as it might sound. There are many factors, including the semantic gap, that can make this task particularly challenging. In this project, we considered a multi-tier object detection system to detect the biological gender and ethnicity based on a face image while taking into consideration some of the semantic gaps. In this report, we described the development and application to the gender & ethnicity detector system. We illustrate the mathematical methods, display the tuning process of building the final models, and elaborate the results. We also discuss further improvements on the current models and make conclusions.

Related Work

Below, we summarized the progress of gender and ethnicity detectors, and widely used detector systems and methods used currently for gender and ethnicity detection using computer vision.

The earliest steps into Facial Recognition by Woody Bledsoe, Helen Chan Wolf, and Charles Bisson who in 1964 and 1965 began their work by using computers to recognize the human face, They were severely hampered by the technology of the era, but it remains an important first step in proving that Facial Recognition was a viable biometric[1].

Vendors like Facebook, Google, Amazon, and many others are constantly working on improving their algorithms for computer vision in general; Facebook has taken an A.I. model approach, known as SEER meaning “self-supervised” where is an ultra-large vision model trained off more than one billion images from Instagram accounts[5]. In addition, there are some vendors such as Amazon and IBM that have provided Law Enforcement with a solution to identify crimes easily by using their detection systems. However, it has had some controversial results as it has been claimed that these models have been used to carry out gender/ethnic biases[3].

In this project, we used a well-known computer vision method-Convolutional Neural Network (CNN), which is an artificial neural network used in image recognition[2]. We also used the TensorFlow application ‘MobileNetV2[8]’ which has been used with pre-trained weights of ImageNet, an image dataset that has been utilized for advancing computer vision and deep learning research [4].

Problem Definition

Building a multi-tier face recognition system, which encompasses a class of techniques and algorithms that can detect the biological gender, and ethnicity of a person. The purpose is to construct a detector system that overcomes the semantic gap, such as:

- Illumination
- Occlusion
- Resolution
- Others

There are five categories in which the identification of ethnicity (White, Black, Asian, Indian, and Others (Hispanic, Latino, and Middle Eastern)) will be taken into consideration, as well as identify two categories of gender (male & female).

Methodology

To build the multi-tier detector system for gender and ethnicity, the following steps were followed:

i. **Data Filtering:**
   a. The **UTKFace**[11] dataset was divided and stored into four parts for smoother training due to memory issues.
   b. Then, the gender and ethnicity datasets were filtered with their corresponding labels in preparation for the model training process.

ii. **Data Preprocessing:**

   a. **preprocess_input** was used to preprocess the images. Then, we reduced the size of each image to the accepted format of size 198x198.
   b. The images were preprocessed as required by the models into the necessary format. This preprocessing involves:
      1. Applying filter
      2. Image conversion into the data array
      3. Transforming Labels to Label binarized with fixed classes

iii. **Data split:**

   a. train_test_split was used to split the data into 80-20 ratio for the training and validation datasets.

iv. **Model Training:**

Model training consist of several parts:

a. **Feature Extraction**: The model was trained on Keras Sequential Model with Conv2D and Pooling function as layers for Sequential model for feature extraction. The CNN layers were used to train the images. For this purpose, we are using the TensorFlow application MobileNetV2 as the base model. A transfer-learning technique was then applied by using the pre-trained ImageNet weights. ImageNet is an image dataset organized according to the WordNet hierarchy[4]. Each node of the hierarchy is depicted by hundreds and thousands of images[4]. Where, MobileNetV2 is a family of neural network architectures designed to support classification, detection, embedding, and segmentation on light devices.

Input	Operator	t	c	n	s
$224^2 \times 3$	conv2d	-	32	1	2
$112^2 \times 32$	bottleneck	1	16	1	1
$112^2 \times 16$	bottleneck	6	24	2	2
$56^2 \times 24$	bottleneck	6	32	3	2
$28^2 \times 32$	bottleneck	6	64	4	2
$14^2 \times 64$	bottleneck	6	96	3	1
$14^2 \times 96$	bottleneck	6	160	3	2
$7^2 \times 160$	bottleneck	6	320	1	1
$7^2 \times 320$	conv2d 1x1	-	1280	1	1
$7^2 \times 1280$	avgpool 7x7	-	-	1	-
$1 \times 1 \times 1280$	conv2d 1x1	-	k	-	

Figure 1: MobileNetV2 architecture

In Figure 1, **t** is an expansion factor, **c** is several output channels, **n** is the repeating number, and **s** is the stride. 3×3 kernels are used for spatial convolution.

Typically, the primary network (width multiplier 1, 224×224), has a computational cost of 300 million multiply-adds and uses 3.4 million parameters (width multiplier is introduced in MobileNetV1). The performance trade-offs are further explored, for input resolutions from 96 to 224, and width multipliers of 0.35 to 1.4[9]. MobileNetV2 is one of the many Keras Applications. These applications are canned architectures with pre-trained weights.

b. **Fully Connected:** Flatten was used to flatten all 3 layers of RGB into the 1D (1 Dimensional) layer. This passed to the Dense layer with an activation function.

c. **Output layer:** SoftMax function as activation function was used in the last output Dense layer. SoftMax ensures that the sum of output probabilities from the Fully Connected Layer is 1. SoftMax function takes a vector of arbitrary real-valued scores and squashes it to a vector of values (between zero and one) that sum to one[12].

d. Two models were trained for the application:

1. **Gender detection model:** This model detects the two biological genders (male & female) based on the input dataset.
2. **Ethnicity detection model:** This model detects five ethnicity classes (White, Black, Asian, Indian, and Others (Hispanic, Latino, Middle Eastern) based on the input dataset.

v. **Real-Time Application:** After the model training was completed, we used the OpenCV library for live camera feed and testing the final models. OpenCV can actively detect multiple faces in an image. These detected faces are resized and converted into array format to pass on to the model for prediction.

Experimental Setting

We used the UTKFace[11] dataset, which is a large-scale face dataset. This dataset:

- Consists of 20k+ face images in the wild (only a single face in one image)
- Provides the correspondingly aligned and cropped faces.
- Provides the corresponding landmarks (68 points)
- Images are labeled by age, gender, and ethnicity where we are only going to use gender and ethnicity:
  - Gender is either 0 (male) or 1 (female)
  - Ethnicity is an integer from 0 to 4, denoting White, Black, Asian, Indian, and Others (Hispanic, Latino, Middle Eastern).

Experimental Results and Analysis

Due to the amount of data (23,712 images), it became impossible to feed that amount of data in low-end computers with only 4 GB of GPU. Therefore, the model training was done in three parts. The following settings were followed for the model training:

- **Common Layers of Model:** The models were not trained using top layers of MobileNetV2 because it has a dense layer of 1000 classes, which is too much

for this purpose. We used our own layers. Below is the structure of the common model layers of our models.

average_pooling2d (AveragePooli	(None, 1, 1, 1280)	0	out_relu[0][0]
flatten (Flatten)	(None, 1280)	0	average_pooling2d[0][0]
dense (Dense)	(None, 128)	163968	flatten[0][0]
dropout (Dropout)	(None, 128)	0	dense[0][0]

- **Output Layers of Model:** The models were not trained using top layers of MobileNetV2. We have used our own layers for the output. Below is the structure of the output model layers of our models.
    - **Gender detection model:**

dense_2 (Dense)	(None, 2)	258	dropout[0][0]

Total params: 2,422,210
Trainable params: 2,388,098
Non-trainable params: 34,112

    - **Ethnicity detection model:**

dense_1 (Dense)	(None, 5)	645	dropout[0][0]

Total params: 2,422,597
Trainable params: 2,388,485
Non-trainable params: 34,112

- **Adam Optimizer:** Both models were compiled using Adam optimizer with initial learning of ‘1e-4’, and decay rate of ‘5.0e-6’.
- **Training Data:** Training data was passed to both model via ImageDataGenerator with the following settings:

rotation_range=20, zoom_range=0.15, width_shift_range=0.2, height_shift_range=0.2, shear_range=0.15, horizontal_flip=True, fill_mode="nearest"

- **Callback functions:** Earlystopping: We introduced Early stopping to stop the model run if 'val_loss' does not improve for 5 consecutive EPOCHS.

Model training result by run and models are shown below:

- Gender detection model training:
  - On the first run, the model was trained on 9,962 images. The first run was done for 20 EPOCHS with an early stop at the 13th EPOCH.

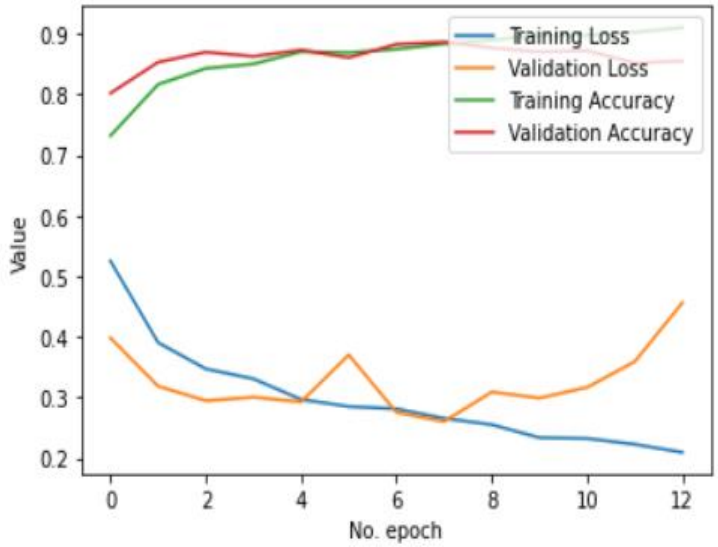


Figure 2: First run Training/Validation Loss and Accuracy Plot for Gender Detection Model

  - On the second run, the model was trained on 6,829 images. Model train was done for 20 EPOCHS with an early stop at 11th EPOCH.

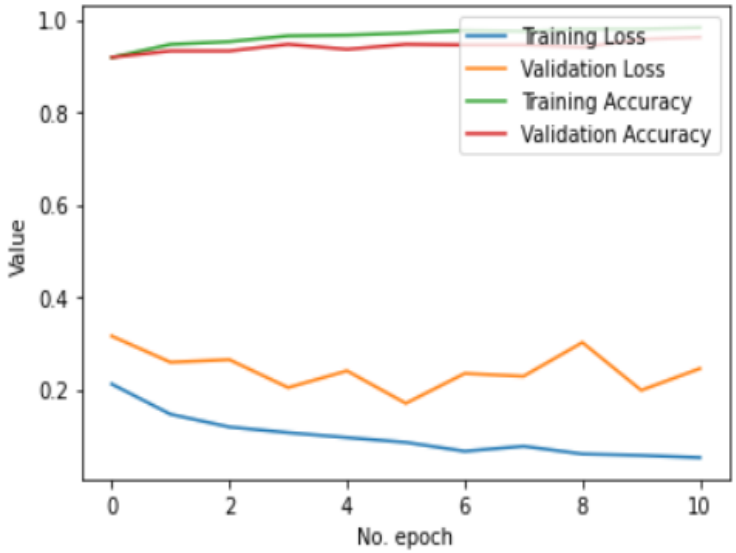


Figure 3: Second run Training/Validation Loss and Accuracy Plot for Gender Detection Model

- On the third run, the model was trained on 6,921 images. Model train was done for 20 EPOCHS with an early stop at 10th EPOCH.

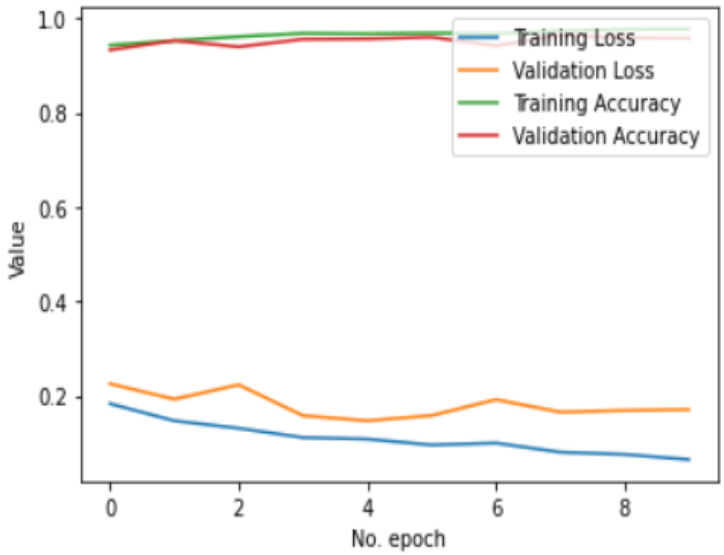
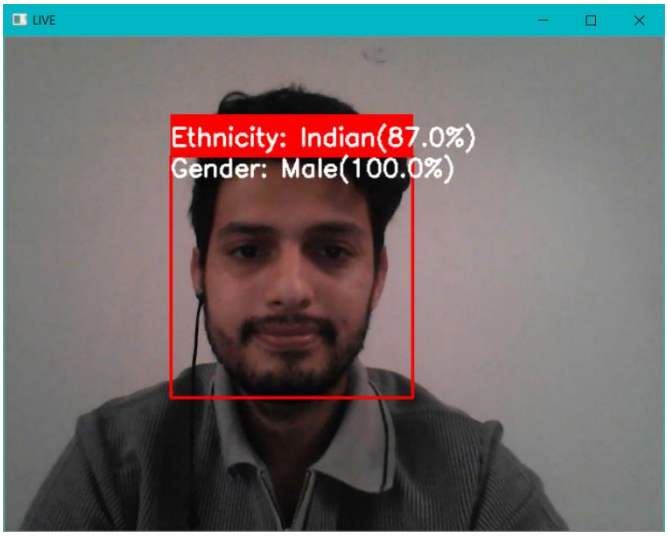


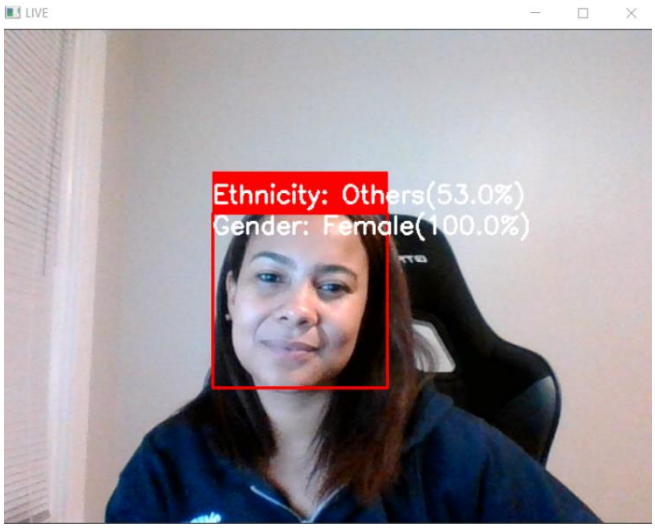
Figure 4: Final run Training/Validation Loss and Accuracy Plot for Gender Detection Model

Final Testing: Real-Time Application



Pardeep Pathania (Ethnicity-Indian, Gender-Male). Model results are as follows:
The model predicted Indian ethnicity with 87% accuracy and predicted male with 100% accuracy.

Vijay Kshetri (Ethnicity-Indian, Gender-Male). Model results are as follows: The model predicted Indian ethnicity with 70.93% accuracy and predicted gender male with 99.83% accuracy.



Yerandy Minaya (Ethnicity-Hispanic, Gender-Female). Model results are as follows:
The model predicted "Others" ethnicity with 53% accuracy and predicted female with 100% accuracy.

Future Work

For future work, we would like to make more precise models and minimize inter-class bias. We believe that there is always a scope for research and redesign of the present model/architecture. In addition, this model could benefit from more images in the training dataset to train the model for improved accuracy.

Conclusion

We were successfully able to design a system using Convolution Neural Networks and the UTKFace dataset. This system is developed in such a way that it can be integrated into any image capturing device which takes an input and then processes it and gives an output with the following accuracies:

- Gender model has accuracy: 88.60%
- Ethnicity model has accuracy: 80.42%
- Both models have high accuracy and low validation loss.
- Models perform well to predict Gender and Ethnicity.

References

[1] "A Brief History of Facial Recognition - NEC New Zealand." *NEC*, 9 Sept. 2020, www.nec.co.nz/market-leadership/publications-media/a-brief-history-of-facial-recognition/. 3

[2] Contributor, TechTarget. "What Is Convolutional Neural Network? - Definition from WhatIs.com." *SearchEnterpriseAI*, TechTarget, 26 Apr. 2018, searchenterpriseai.techtarget.com/definition/convolutional-neural-network. 3

[3] "George Floyd: Amazon Bans Police Use of Facial Recognition Tech." *BBC News*, BBC, 11 June 2020, www.bbc.com/news/business-52989128#:~:text=Technology%20giant%20Amazon%20has%20banned,recognition%20software%20for%20a%20year.&text=Amazon%20said%20the%20suspension%20of,how%20the%20technology%20is%20employed. 3

[4] *ImageNet*, www.image-net.org/. 4, 5

[5] Kahn, Jeremy. "Facebook Claims Computer Vision Breakthrough with Instagram-Trained A.I." *Fortune*, Fortune, 4 Mar. 2021, fortune.com/2021/03/04/facebook-says-its-new-a-i-that-learns-without-labelled-data-represents-a-big-leap-forward-for-computer-vision/. 3

[6] Mishra, Abhishek. "Machine Learning in the AWS Cloud: Add Intellegence to Applications with Amazon SageMaker and Amazon Rekognition." *Amazon*, John Wiley & Sons, Inc., 2019, aws.amazon.com/rekognition/?blog-cards.sort-by=item.additionalFields.createdDate&blog-cards.sort-order=desc.

[7] Somdip DeySomdip Dey 3, and Dmytro PrylipkoDmytro Prylipko 3. "Create CNN Model Architecture Diagram in Keras." *Stack Overflow*, 1 Nov. 1967, stackoverflow.com/questions/54943307/create-cnn-model-architecture-diagram-in-keras.

[8] "Tf.keras.applications.MobileNetV2 : TensorFlow Core v2.4.1." *TensorFlow*, www.tensorflow.org/api_docs/python/tf/keras/applications/MobileNetV2. 3

[9] Tsang, Sik-Ho. “Review: MobileNetV2 - Light Weight Model (Image Classification).” *Medium*, Towards Data Science, 1 Aug. 2019, towardsdatascience.com/review-mobilenetv2-light-weight-model-image-classification-8febb490e61c. 5

[10] Ujjwalkarn. “An Intuitive Explanation of Convolutional Neural Networks.” *The Data Science Blog*, 29 May 2017, ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/.

[11] *UTKFace*, susanqq.github.io/UTKFace/. 4, 6

[12] Gudikandula, Purnasai. “A Beginner Intro to Neural Networks.” *Medium*, Medium, 24 Mar. 2019, purnasaigudikandula.medium.com/a-beginner-intro-to-neural-networks-543267bda3c8. 5