Simultaneous Detection of Face Attributes via Multi-Task Learning

Student Name: Himanshu Sundriyal

IIIT-D-MTech-CS-MT18074

June, 2020

Indraprastha Institute of Information Technology
New Delhi

Thesis Advisors
Dr. Mayank Vatsa
Dr. Richa Singh

Submitted in partial fulfilment of the requirements for the Degree of M.Tech. in Computer Science

©2020 IIIT-D-MTech-CS-MT18074
All rights reserved

Certificate

This is to certify that the thesis titled **"Simultaneous Detection of Face Attributes via Multi-Task Learning"** submitted by **Himanshu Sundriyal** for the partial fulfillment of the requirements for the degree of *Master of Technology* in *Computer Science & Engineering* is a record of the bonafide work carried out by him under our guidance and supervision at Indraprastha Institute of Information Technology, Delhi. This work has not been submitted anywhere else for the reward of any other degree.

Dr. Mayank Vatsa
Indian Institute of Technology Jodhpur

Dr. Richa Singh
Indian Institute of Technology Jodhpur

Abstract

Identifying face attributes is an ongoing problem of research which is used in bio-metrics, surveillance etc. In past, researchers have proposed methods which predict single facial attribute at a time. Real-time applications need to predict multiple attributes simultaneously to increase their efficiency in real world. Recently, researchers have started to use Multi-Task learning to learn multiple facial attributes simultaneously and leverage the correlations among the tasks. Using MTL, related tasks can generalise better as compared to single task learning and it requires only one model to predict multiple features thus making it more efficient for testing in real-time.

In this research, we have proposed a Multi-Task learning architecture which simultaneously predicts gender, age and race of given input facial image. The model outperforms the methods which learn only single task at a time. We have used CelebA and UTKFace dataset to assess the effectiveness of the proposed MTL architecture. We got best results for UTKFace dataset when we used SE-ResNet-50-128D as pretrained model with gender recognition accuracy equal to 98.47 %, race prediction accuracy equal to 87.56 % , and age Mean Squared Error equal to 4.68 years. We got best results for CelebA dataset when we used SE-ResNet-50-128D as pretrained model with gender recognition accuracy equal to 98.82 % and age prediction accuracy equal to 87.53 %. We compared these results against previous works on CelebA and UTKFace datasets. Proposed architecture outperformed current state-of-the-art architectures in most of the cases.

Keywords : Multi-Task Learning, Transfer Learning, Cross Entropy Loss, Mean Squared Error, Convolutional Neural Network

Acknowledgments

I would like to express my sincere gratitude to my advisors Dr. Mayank Vatsa and Dr. Richa Singh for their continuous support and guidance throughout my thesis work. They were always available for help and guidance whenever I got stuck in the research work and gave apt suggestions to proceed towards the correct direction. I would also want to thank my mentor Rohit Keshari for his never-ending support during my thesis work. He was always available for help either in-person or via phone calls and gave theoretical as well as practical suggestions whenever I was in some doubt. I would also like to thank M.Tech Academic Department for their constant support and help. Last but not the least, I would like to thank my family for their endless love and support which helped me to accomplish this thesis work.

Contents

1 Introduction **1**

1.1 Research Contributions 4

1.2 Organisation of Thesis Report 5

2 Related Work **6**

3 Proposed Multi-Task Learning Architecture **10**

3.1 Data Preprocessing 11

3.2 Proposed MTL Network 11

3.3 Task Specific Loss Functions 13

3.3.1 Loss Function for Gender Classification Task 13

3.3.2 Loss Function for Age Estimation Task 13

3.3.3 Loss Function for Race Classification Task 14

3.4 Proposed Weighted Loss Function 15

3.5 Proposed Early Freeze Algorithm for MTL 16

3.5.1 Early Freeze Algorithm 16

3.6 Implementation Detail 17
3.6.1 Tools and Framework 17
3.6.2 Hyperparameters 17

4 Experiments, Results and Observations 19

4.1 Datasets 19
4.2 Dataset Experimental Protocol 22
4.3 Pretrained Models for Facial Feature Vector Extraction 23
4.4 Results and Observations 24
4.4.1 Comparison with Baselines on UTKFace Dataset 24
4.4.2 Comparison with Baselines on CelebA Dataset for Different Pretrained Models 27
4.4.3 Comparison of Gender Accuracy on CelebA Dataset 28
4.4.4 Comparison of Age Prediction Accuracy on CelebA Dataset 30
4.4.5 Comparison of Age Mean Absolute Error (MAE) on UTKFace Dataset 31
4.4.6 Misclassified Samples from MTL+SEResnet on CelebA Dataset . . . 32
4.4.7 Results on Real-World Images 33

5 Analysis of Proposed MTL Method 35

5.1 Performance Improvement with Proposed Weighted Loss 35
5.1.1 CelebA Dataset 35
5.1.2 UTKFace Dataset 36
5.2 Performance Improvement with Proposed Early-Freeze Method 37

5.3 Mitigation of Gender and Age Bias due to Multi-Task Learning on CelebA Dataset 39

5.4 Hyperparameter Tuning for Regularisation Hyperparameter (λ) in the Proposed Weight Loss 41

6 Conclusions and Future Work **42**

Bibliography **46**

List of Tables

2.1 Summary of MTL Methods Used for Face Attributes Detection 8

2.2 Summary of Recent Research on MTL Architectures 9

3.1 Tools and Frameworks Used 17

3.2 Hyperparameter Values 18

4.1 UTKFace Dataset Distribution According to Race and Gender Classes . . . 20

4.2 CelebA Dataset Distribution According to Gender and Age Classes 21

4.3 Dataset Split in UTKFace Dataset 22

4.4 Dataset Split in CelebA Dataset 23

4.5 Pretrained Models Trained for Face Detection Task 24

4.6 Comparison of Methods when Pretrained Model is LightCNN on UTKFace 26

4.7 Comparison of Methods when Pretrained Model is Sphereface on UTKFace 26

4.8 Comparison of Methods when Pretrained Model is Inception-ResNet (Facenet) on UTKFace 26

4.9 Comparison of Methods when Pretrained Model is ResNet-50 on UTKFace 26

4.10 Comparison of Methods when Pretrained Model is SE-ResNet-50-128D on UTKFace 26

4.11 Comparison of Methods when Pretrained Model is LightCNN on CelebA Dataset 27

4.12 Comparison of Methods when Pretrained Model is Sphereface on CelebA Dataset 27

4.13 Comparison of Methods when Pretrained Model is Inception-ResNet on CelebA 27

4.14 Comparison of Methods when Pretrained Model is ResNet-50 on CelebA . . 28

4.15 Comparison of Methods when Pretrained Model is SE-ResNet-50-128D on CelebA 28

4.16 Comparison of MAE (in years) for Age Estimation Task on UTKFace Dataset 31

5.1 Table Showing Improvement in CelebA Results when Proposed Weighted Loss was Used Instead of Average Loss and Pretrained Model Used is SE-ResNet-50 35

5.2 Table Showing Improvement in CelebA Results when Proposed Weighted Loss was Used Instead of Average Loss and Pretrained Model Used is ResNet-50 36

5.3 Table Showing Improvement in UTKFace Results when Proposed Weighted Loss was Used Instead of Average Loss and Pretrained Model Used is SE-ResNet-50 36

5.4 Table Showing Improvement in UTKFace Results when Proposed Early Freezing was Used and Pretrained Model Used is SE-ResNet-50 39

5.5 Table Showing Results for Different Values of Hyperparameter λ and Pre-trained Model Used is SE-ResNet-50 41

List of Figures

1.1 Multi-Task Network Detecting Facial Attributes Simultaneously 2

1.2 Output from Proposed MTL Network Trained on CelebA Dataset to Predict Age (Young or Old) and Gender (Male or Female) 3

1.3 Output from Proposed MTL Network Trained on UTKFace Dataset to Predict Age, Race and Gender from Given Image 3

3.1 Steps Involved in Proposed Algorithm 11

3.2 Proposed MTL Architecture 12

4.1 UTKFace Dataset Age Distribution per Gender Category (Male and Female) 20

4.2 UTKFace Dataset Samples 21

4.3 CelebA Dataset Samples 22

4.4 Gender Prediction Single Task Model 25

4.5 Age Prediction Single Task Model 25

4.6 Race Prediction Single Task Model 25

4.7 Comparison of Gender Recognition Accuracy (in Percentage) on CelebA Dataset 29

4.8 Comparison of Age Prediction Accuracy (in Percentage) on CelebA 30

4.9 Misclassified Images in Age Prediction Using MTL+SEResnet Model . . . 32

4.10 Misclassified Images in Gender Prediction Using MTL+SEResnet Model . 32

4.11 Testing Framework for Real-World Images 33

4.12 Simultaneous Face Attribute Detection from Real-World Images using MTL Model Trained on CelebA Dataset 34

4.13 Simultaneous Face Attribute Detection from Real-World Images using MTL Model Trained on UTKFace Dataset 34

5.1 Loss vs Iterations Graphs when Early Freeze Method used 37

5.2 Loss vs Iterations Graphs when Early Freeze Method is not used 38

5.3 Confusion Matrix for Single Task Neural Network Using ResNet-50 Pre-trained Model on CelebA Dataset. Confusion Matrix Clearly Depicts Bias Given to Young Task than the Old Task 40

5.4 Confusion Matrix for Proposed MTL Network Task Using ResNet-50 Pre-trained Model on CelebA Dataset 40

Chapter 1

Introduction

Multi-Task Learning (MTL) [2, 3] aims to improve the performance of multiple related tasks by jointly learning these tasks. When tasks at hand are correlated, we can leverage the relationship among the tasks so as to improve the efficiency of all the tasks. Since we are learning all tasks simultaneously, we are no longer required to train n models for n different tasks, thus reducing the training time as well. MTL also has a regularisation effect when implemented for related tasks, as it increases the generalisation efficiency of all the tasks and reduces chances of overfitting.

MTL can be thought of as learning to play the Guitar and a Ukulele at the same time. Both are different tasks, but since both are musical stringed instruments, they have some relationship like similar chord patterns etc. The learning involved in playing the Guitar can be used to learn a Ukulele and vice-versa. In simultaneous face attribute detection for age, race, and gender, these relationships are not directly seen and are figured out by the part of the deep learning model shared among the tasks.

While MTL seems to be similar to Transfer learning technique, these two have a significant difference in terms of learning multiple tasks. In case of transfer learning, we have a pretrained model which is trained on some task A; then we use this model by retraining some layers based on task B to increase the efficiency of the model on task B. Thus in transfer learning we have a task which is important for us, and we want to get better results on this task by using another task. In MTL we also have multiple tasks, but there is no discrimination among the task, rather we want all of

the tasks to do well by sharing some information among them [24].

The problem of identifying face attributes is an ongoing problem of research which is used in bio-metrics, surveillance etc. Today Deep Convolution Neural Network (CNN) solves the problem of face verification and detection with very high efficiency but gender classification, age estimation, race classification, pose estimation etc. are still challenging problems due to different resolution conditions, illumination conditions, side pose of faces etc. Previously researchers have proposed many methods which are used to predict or classify one individual attribute at the time such as age, gender etc looking at a face image. Identifying facial attributes using different models make it difficult to use in real-time face attributes prediction system as different models take separate storage space and take time to predict facial attributes one by one. For a real-time system, if there is only one model which does all the facial attributes detection simultaneously, then it takes lesser space and time.

More recently, researchers have tried to apply Multi-Task Learning to leverage the correlation among the tasks at hand. Using MTL, one can decrease the training time to train multiple models individually and also can hope to extract some information among the tasks which are used by another task to improve its performance. Using MTL thus helps to combine the tasks and provides a single model which gives all the required attributes values at the same time with better performance.

Below result is expected from the proposed MTL model to predict age, race, and gender at the same time.

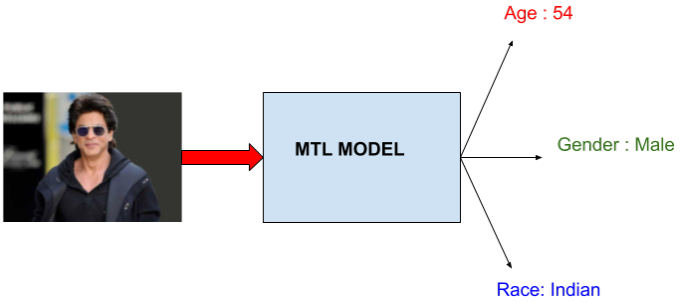


Figure 1.1: Multi-Task Network Detecting Facial Attributes Simultaneously

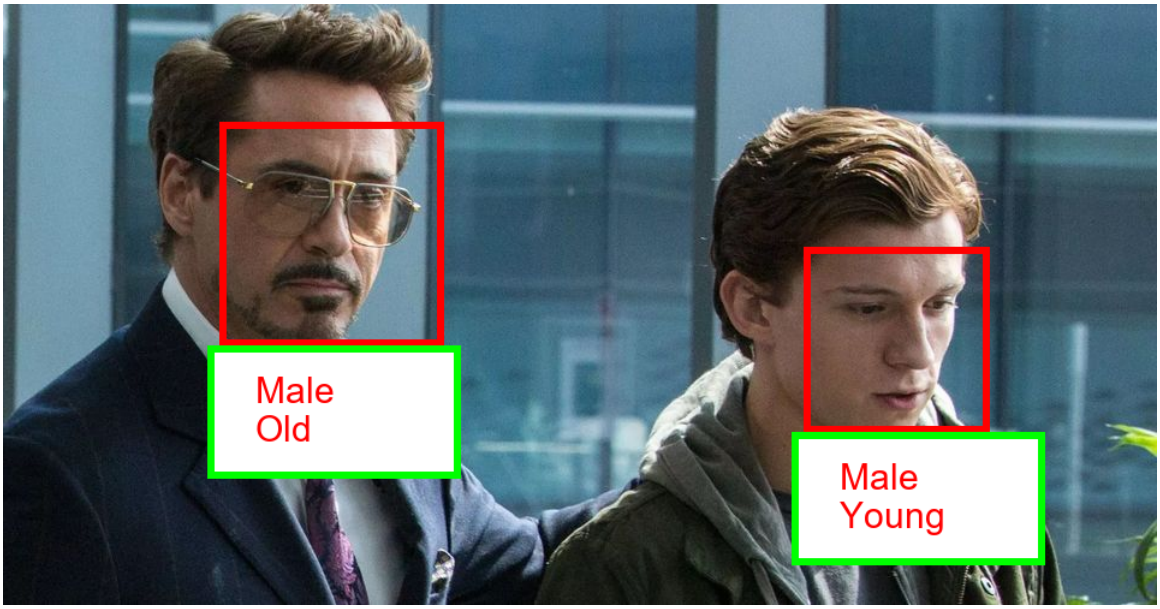


Figure 1.2: Output from Proposed MTL Network Trained on CelebA Dataset to Predict Age (Young or Old) and Gender (Male or Female)

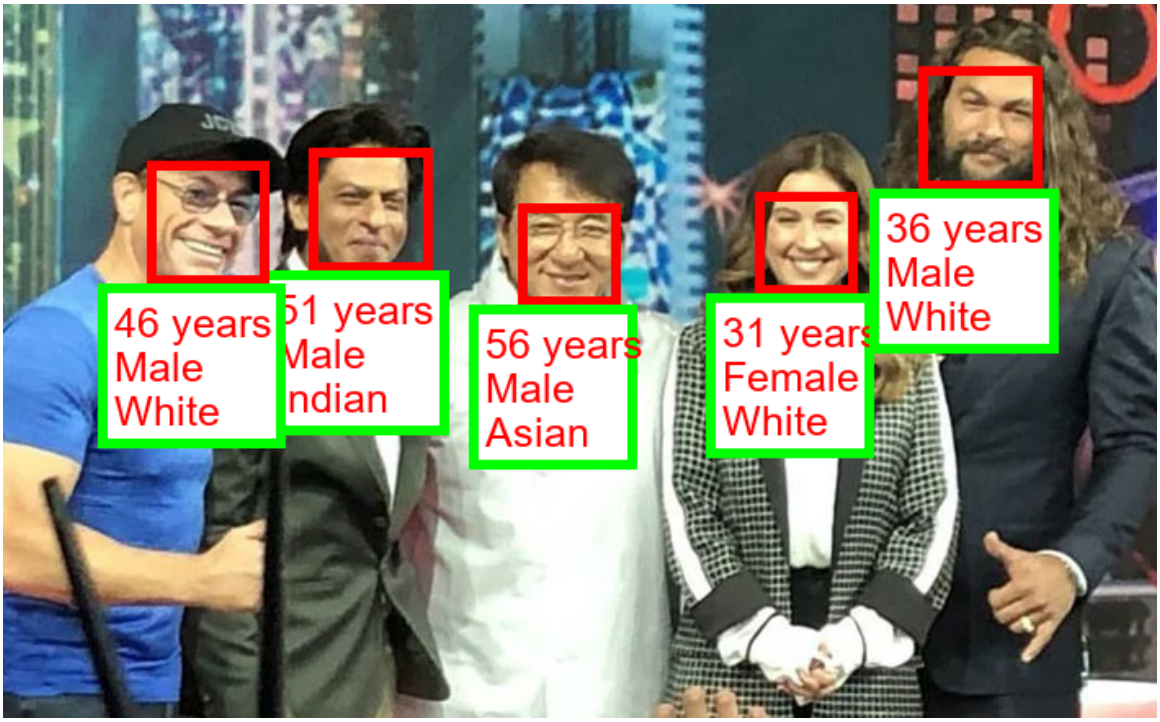


Figure 1.3: Output from Proposed MTL Network Trained on UTKFace Dataset to Predict Age, Race and Gender from Given Image

1.1 Research Contributions

In this work, we proposed a Multi-Task Learning (MTL) network which has convolution and fully connected layers shared layers among the different tasks followed by task-specific fully connected layers for different tasks. So the model allows learning the common representations in the lower layers and learning task-specific work at higher levels of the network [21].
We have used UTKFace [23] and CelebA [9] dataset to assess the efficiency of the proposed MTL algorithm. MTL helps to reduce the training time and increase the generalisation performance and the robustness of the overall model.
Key highlights of the research are following:

1. **Multi-Task Learning (MTL) Architecture**
   We propose an MTL Architecture which simultaneously predicts gender, age and race of the given face image. This proposed MTL architecture outperforms the single task CNN networks. We have also compared our results with current state-of-the-art methods. The proposed model outperforms the current state-of-the-art model in most of the cases.

2. **Learning of weights given to the task-specific losses**
   While in previous works done by researchers in MTL, the task-specific losses are combined by using a weighted sum of losses, where the weight given to each of the loss is assigned empirically.
   In the proposed method, all the losses obtained from gender, age and race tasks are still combined using weighted sum, but the weights given to each of the loss are learned itself by the network. The network gives random weights initially to these losses with the sum of all weights equal to 1. The MTL network then learns theses loss weights in the training process. So there is no guessing involved in assigning weights to the losses, rather the network learns the best weights for the losses while training. We observed that the overall performance of the model is increased as compared to the case when weights were given to losses empirically.

3. **Early-Freezing**
   In MTL, tasks are learned simultaneously, so some tasks can converge faster than other tasks. When the converged tasks stop learning, the non converged

tasks affect the shared parameters of the model, thus decreasing the performance of the already converged tasks. To avoid this problem, we propose novel *"Early Freeze"* approach to freeze the shared layer parameters from training as soon as one of the tasks converge. This helps in not decreasing the performance of the earlier converged tasks.

1.2 Organisation of Thesis Report

The organisation of the thesis report is as follows. Chapter 2 describes the previously related works in the field of facial attribute detection and Multi-Task learning. Chapter 3 discusses the proposed Multi-Task network and network pipeline. This chapter also describes the proposed loss function for the MTL network, proposed Early Freeze method and Implementation details. Chapter 4 discusses about the datasets used, experiments on these datasets and their respective observations. Chapter 5 describes the analysis of proposed MTL architecture and proposed weighted loss function. Finally, Chapter 6 discusses the summary of the proposed MTL method, conclusions and future work.

Chapter 2

Related Work

Rich Caruana [2, 3] published the first work on Multi-Task Learning. The work demonstrated that MTL could be used in real-life domains. Multi-Task learning can be applied in real-life tasks, e.g. Computer Vision tasks, Natural Language Processing tasks etc. In Computer Vision, till now, many researchers have used MTL technique to increase the generalisation performance of the related tasks. In general, there is sharing of the convolutional and/or fully connected layers among the tasks and then separate fully connected (FC) layers for each of the different tasks.

MTL can also be used in predicting the facial attributes like gender, age, race, pose, smile etc. Instead of training separate models to predict age, gender, race etc. MTL aims to train a single model that does simultaneous learning of different face attributes and tries to leverage the correlation among these tasks. Zhang et al. [24] gave survey for Multi-Task Learning. They classified different MTL methods into categories, namely task-clustering, task-relation learning, low-rank, feature learning, and decomposition approaches. Following are some previous works for facial attribute detection using MTL.

- Levi et al. [7] used a deep-convolutional network to find the representations of the face image, to simultaneously detect gender and age attributes of the face. They used Adience dataset for evaluation of gender and age predicting capability of the MTL model.
- Yi et al. [20] used Canonical Correlation Analysis method (CCA) to simultane-

ously detect race, age, and gender for a facial image and showed that multi-task regression method based on CCA gives better results than partial least square (PLS) method.

- Ranjan R et al. [14] proposed HyperFace, which detects a face, localises facial landmarks, estimates face pose and predict the gender of the given face at the same time. It used the shared CNN layers to identify the shared correlation among the tasks and then separate task-related layers for each of the tasks. Further, they also fused CNN hyperfeatures. They showed that by analysing local and overall knowledge of the face image, it improved the performance of the tasks as compared to when tasks were learned individually.

- Ehrlich et al. [4] proposed the MTL model to classify attributes of the face which learns representations which are shared among the different tasks. They used the Restricted Boltzmann Machine (RBM) model to learn shared representation followed by MTL framework and named it as multi-task RBM (MT-RBM). MT-RBM simultaneously detected the face attributes with improved performance.

- Ranjan et al. [15] proposed "All-In-One Convolutional Neural Network for Face Analysis" which performed gender prediction, smile prediction, detection and alignment of face concurrently using MTL. The model also regularised parameters of the network using MTL.

- Wang et al. [18] proposed Deep MTL (DMTL) which learned facial feature representation using correlation and heterogeneity and simultaneously predicted facial attributes are belonging to a given face. The proposed model used CNN to learn shared features across all the facial tasks and then task-specific layers to learn features corresponding to different tasks. It also introduced LFW+ dataset which extended the LFW dataset, for simultaneous heterogeneous attribute learning.

- Liu et al. [8] proposed joint MTL using CNN to simultaneously predict facial attributes in the wild. They predicted all the 40 attributes in CelebA and LFWA using MTL.

Paper	Author	Tasks	Dataset	Publication Year
Age Estimation by Multi-scale Convolutional Network	Yi et al.	Age,Gender,Ethnicity	MORPH Album 2	2014
Age and Gender Classification using Convolutional Neural Networks	Levi et al.	Gender,Age	Audience	2015
Facial Attributes Classification using Multi-Task Representation Learning	Ehrlich et al.	Gender, Smile	MTFL, ChaLearn, CelebA	2016
HyperFace: A Deep Multi-Task LearningFramework for Face Detection, Landmark Localization, Pose Estimation, and Gender Recognition	Ranjan et al.	Face Detection, Landmark Localization, Pose Estimation, and Gender Recognition	AFLW, IBUG, CelebA	2017
An All-In-One Convolutional Neural Network for Face Analysis	Ranjan et al.	Gender,Age, Smile, Pose, face Detection and Verification, Landmark Localisation	AFLW, AFW, CelebA, ChaLearn, LAP2015, FGNet, IJB-A	2017
Deep Multi-Task Learning for Joint Prediction of Heterogeneous Face Attributes	Wang et al.	Gender,Age, Race	MORPH II, CelebA, LFWA	2017
Deep Learning Face Attributes in the Wild	Liu et al.	Gender, Age ,Smile	CelebA, and LFWA	2017
Multi-task Learning of Cascaded CNN for Facial Attribute Classification	Zhuang et al.	Gender, Age, Smile	CelebA, LFWA	2018
Task Relation Networks	Li et al.	Gender, Age, Smile	CelebA, LFWA	2019
Deep Multi-task Multi-label CNN for Effective Facial Attribute Classification	Mao et al.	Gender, Age, Smile	CelebA, LFWA	2020

Table 2.1: Summary of MTL Methods Used for Face Attributes Detection

- Mao et al. [12] jointly learned face attributes and localisation using MTL. They divided face attributes into two groups, subject group and objective group and trained two different networks for these two groups. They compared the results on LFWA and CelebA dataset.

Apart from using MTL in learning facial attributes simultaneously, MTL is also used in other fields of computer vision, natural language processing, speech recognition etc. Below is some of the recent researches which proposed novel MTL architectures.

- Research by Misra et al. [13] used different model architectures for different tasks and then there is stitching among the layers of different task-specific layers.
- Lu et al. [11] proposed a network that uses a greedy approach to widen the network while training but as greedy approaches look at local nodes, it is not guaranteed that the model created using widening is optimal.
- Long et al. [10] proposed Deep Relationship Network. They introduced matrix priors among the task specific fully connected layers to learn commonalities among the tasks.
- Ruder et al. [16, 17] proposed an MTL model which combines the stitching architecture by [13] with hard parameter sharing approach.

Paper	**Author**	**Publication Year**
Cross-Stitch Networks for Multi-Task Learning	Misra et al.	2016
Representation Learning Using Multi-Task Deep Neural Networks for Semantic Classification and Information Retrieval	Lu et al.	2017
Learning Multiple Tasks with Multilinear Relationship Networks	Long et al.	2017
Latent Multi-Task Architecture Learning	Ruder et al.	2019

Table 2.2: Summary of Recent Research on MTL Architectures

Chapter 3

Proposed Multi-Task Learning Architecture

We propose a Multi-Task Network in which all the tasks are learned simultaneously by the deep learning model. The deep learning model consists of shared CNN layers followed by task specific layers. By sharing the lower layers among the tasks, we can leverage the relationship between the tasks so as to improve the efficiency of all of the tasks.

In general, the cropped face image is required for most of the facial attributes detection task [15]. Thus the face images are first passed through the face detection and alignment framework which detects the face in the image and crops as well as aligns that face region. The cropped face images are then passed through the proposed MTL network. The MTL network consists of a model which is pretrained on face detection task followed MTL layers. The MTL layers have some layers shared among the tasks and are followed by task specific layers (Figure 3.2). The face image is first passed through the pretrained model which extracts a feature vector and then this feature vector is passed through the proposed MTL layers.

We have performed experiments on two datasets, namely UTKFace [23] and CelebA dataset [9]. In the case of UTKFace dataset, initial layers are shared among the tasks of age, race, and gender recognition, and the higher layers are task-specific layers for age, race and gender. While in the case of CelebA dataset, initial layers are shared among the tasks of age and gender recognition, and the higher layers are task-specific layers for age and gender.

3.1 Data Preprocessing

The RGB images given in the dataset contains face images. This face image is detected and extracted from the RGB image to separate the face from the background. Since we want to find out age, race and gender by using facial features, the background feature is not required. We have used face detection algorithm provided by Dlib-ml library [5]. Before giving face images to the deep learning model, the images per batch are re-scaled according to the pretrained model used, e.g for ResNet-50 model pretrained on VggFace2 dataset, face images are rescaled to 200×200.

Figure 3.1: Steps Involved in Proposed Algorithm

3.2 Proposed MTL Network

The proposed Multi-Task learning Network is a single deep learning based network that simultaneously predicts the age, race, and gender of the given facial image. After the preprocessing step, facial image is of size according to the pretrained network, as explained above.
The proposed architecture is shown below in Figure 3.2. We first pass the face image to the network, which is pretrained for face detection task. We removed the last layer of the face detection pretrained network, thus this pretrained network gives a feature vector for every face image provided.

We have used a pretrained network as it gives a head start to the face attribute prediction task since the CNN filters of the pretrained network have facial information e.g some filters trained to detect eyes, nose etc, while some filters to detect other facial features like wrinkles, colour etc [21]. Thus the feature vector provided by the pretrained network is much more informative. The pretrained network also act as the

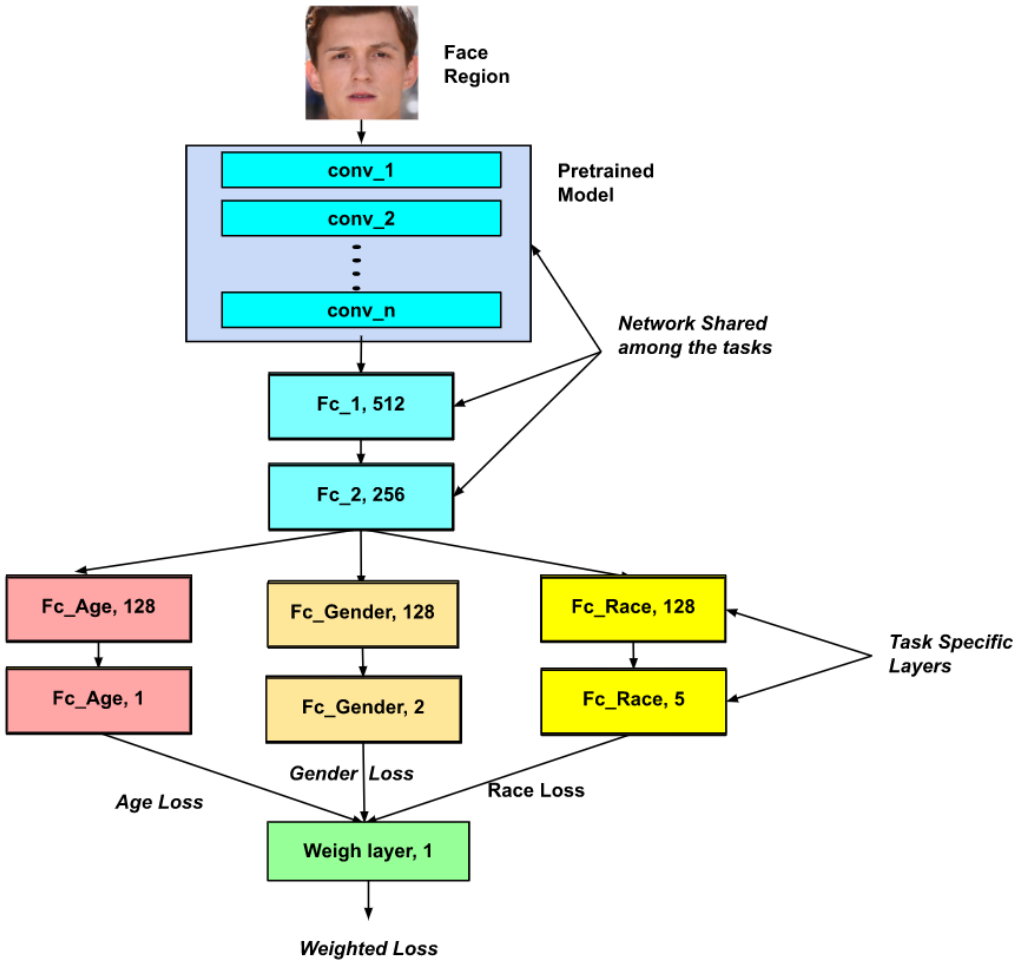


Figure 3.2: Proposed MTL Architecture

shared network for all the tasks.

The produced feature vector from the pretrained network is passed through the proposed MTL layers. Note that the proposed architecture as shown in Figure 3.2 is for UTKFace dataset. Since age is from 0-116 years, age task is considered as a regression problem. For CelebA dataset, age is a classification problem as age has only two labels Young and Old, so for this dataset, the last fully connected layer for age is 128x2. Other layers are the same for both the dataset.

The shared layers of the MTL architecture shown in cyan are followed by Leaky ReLU activation unit and then Batch Normalisation layers. The penultimate task specific layers are followed by dropout layers and Leaky ReLU activation layers. The bottom-

most task specific layers do not have dropout and ReLU layers. The age, gender and race outputs are then used to compute the corresponding losses. The age, race and gender losses are then combined, as explained in Section 3.4. This combined loss is then used in back-propagation for updating parameters of the network.

3.3 Task Specific Loss Functions

Let I be the given input face image. For this image I, we have age label as A, gender label as G and race label as R. This image I is passed to the pretrained network which produces a feature vector which is then passed to the MTL network to produce age loss L_A, race loss L_R, and gender loss L_G. Below is the explanation of tasks and their corresponding loss functions.

3.3.1 Loss Function for Gender Classification Task

Gender Classification is a binary classification problem. It has two labels, namely Male and Female. We have used binary cross entropy loss function for Gender Loss, L_G as shown below

$$L_G = -y_G \times log(p_G) - (1 - y_G) \times (1 - log(p_G)) \tag{3.1}$$

where, y_G is 1 for Male and 0 for Female and p_G is the probability with which model predicts input image I is a Male.

3.3.2 Loss Function for Age Estimation Task

- In UTKFace Dataset, Age is given as integer ranging from 0 to 116 years, so age estimation task is considered as regression task. Loss function for this age estimation task is Mean Squared Error. Squared error for a given Image I is

$$L_A = 0.5 \times (y_A - p_A)^2 \tag{3.2}$$

  where, y_A is the target age of I and p_A is the predicted age of I. This squared error is then summed for the batch and this sum divided by the batch size results

in mean squared error for the batch.
For Age Mean Squared Error (MSE), we have used normalisation technique to bring Age MSE in the range of Gender and Race Losses. The Age ranging from 21 to 60 years is labeled as 0 to 40. We divided the target and predicted ages by 40, to bring labels for Age task between 0 to 1. This maps the Age MSE Loss in the range of Race and Gender Loss.

- In CelebA dataset, Age has only 2 labels Young and Old. So this is a binary classification task and binary cross entropy loss function is taken as Age loss and is given by

$$L_A = -y_A \times log(p_A) - (1 - y_A) \times (1 - log(p_A)) \tag{3.3}$$

where y_A is 1 for Young and 0 for Old and p_A is the probability with which model predicts input I is a Young.

3.3.3 Loss Function for Race Classification Task

- In UTKFace Dataset, 5 Race classes are given, namely White, Black, Asian, Indian and others. This task is a multi-class classification problem and its loss is given by

$$L_R = -\sum_{i=1}^{5} y_{R_i} \times log(p_{R_i}) \tag{3.4}$$

where, y_{R_i} is ground truth and p_{R_i} is the probability score for each race class i, and i $\in [1, 5]$

- In CelebA dataset, there is no Race class, so this loss component L_R is taken as 0.

3.4 Proposed Weighted Loss Function

- The previous works on MTL combined the losses of various tasks according to some weight,w given to each loss, where weights are assigned empirically.
- **Why this is not efficient way?**
  The combined loss is just the weighted sum and so change in a total loss is just the weighted sum of change in all the task specific loss. So a loss term which is more in magnitude gets more weight-age during backpropagation. By giving weights empirically, it is possible that giving a larger weight to a task-specific loss gives that task a bias which helps this task to converge better and other tasks with lower loss weights might not converge better. Another possibility is that if we give equal weights to all losses, still a loss which is greater in magnitude gets more bias towards convergence (Refer to Analysis of proposed weighted loss function 5.1). This problem is similar to the problem of unbalanced dataset i.e. the class with a more significant number of training samples gets more bias.
- In the proposed method, all the losses obtained are passed through a Linear Layer with no bias which acts as a **"Loss Weight Layer"** and the output of this layer is combined Loss.

$$L_{Total} = \sum_{i=1}^{t} w_i \times L_i + \lambda \times \|W\|^2 \tag{3.5}$$

  where $\sum_{i=1}^{t} w_i = 1$ and $i = 1, 2, ...t$, where t is number of tasks.
- These weights are set randomly initially, and the network apart from learning the feature and task representations also learn these weights given to the losses while training during backpropagation.
- The loss function has an additional term $\lambda \times \|W\|^2$ which helps in regularisation of weights applied to these losses. Here λ is the regularisation parameter for the weighted loss and $\|W\|$ is L2 normalisation of W. It is one of the hyperparameter and we have optimised its value that gives best performance to the model. The optimal value of λ is found to be 0.1.
- *Thus, there is no guessing involved in assigning weights to the losses.* ***The***

network itself learns the best weights that should be given to the losses so that overall performance of the model is increased

3.5 Proposed Early Freeze Algorithm for MTL

- Since all the tasks are learned simultaneously, there is a high possibility that some of the tasks converge faster than the others. When some of tasks converge, and we are still learning the shared layers, the performance of already converged tasks decreases as the parameters of the shared layers are now only changed according to the tasks which are not yet converged.
- To prevent this performance drop of the already converged task, we propose a novel "Early Freezing while training" approach to freeze the shared layers as soon as one of the tasks converge. This helps in avoiding the performance drop of the earlier converged tasks and we can still learn the other tasks using the task-specific layers only.

3.5.1 Early Freeze Algorithm

Algorithm 1: Early Freeze Algorithm

Input: Multi-Task Model, Tasks
Parameters : (P) Patience parameter, (K) No. of tasks in MTL model
Output: MTL model is Early Freezeed

```
// Initialize a Counter array with K zeros
Counter = [0,0,0.....,0_k]
for i ← 1 to K do
    task = Tasks[i]
    if val_loss[i] > minimum_val_loss[i] then
        Counter[i] = Counter[i] + 1
    else
        minimum_val_loss[i] = val_loss[i]
    if Counter[i] >= P then
        Freeze shared layers of MTL model and layers corresponding to task i
        Restore the model parameters to the state when Counter[i] = 0
        break
```

The proposed Early Freeze algorithm is presented above (Algorithm 1). The algorithm monitors validation loss for each task. When a task gives a poor performance on the validation set as compared to minimum validation loss till now for that task(line 5 of Algorithm 1), we wait for some number of epochs (line 7 of Algorithm 1) to see if the validation loss decreases. If this does not happen, we stop the training of that task and freeze the parameters of the pretrained model and the shared layers (line 8 of Algorithm 1), and revert the model state to earlier epoch when that task performed best (line 9 of Algorithm 1). Then we start the training of the remaining tasks on the task-specific layers only, thus not effecting the already learned shared parameters. This helps the model to still learn for the non converged tasks while not hurting performance of the already converged tasks.

3.6 Implementation Detail

3.6.1 Tools and Framework

Following are the tools which are used to implement the MTL model. The tools with the version are given below, so that one can reproduce the results.

Tool	Version
Python	3.7.4
Pytorch	1.5.0
Torchvision	0.6.0

Table 3.1: Tools and Frameworks Used

3.6.2 Hyperparameters

Tuning of hyperparameters is very important for training deep learning models. We have used different pretrained models, namely LightCNN, Sphereface, Inception-ResNet, ResNet-50 and SE-ResNet-50-128D (Table 4.5). Finding the optimal values of the hyperparameters was crucial for these pretranied models. Below is the list of hyperparameters and their values that gave best results for the given MTL task at hand.

Hyperparameter	**Value**
Learning rate, α	$3 \times 10^{-}5$
Optimiser	Adam
Weight Decay	$10^{-}5$
$\beta 1$	0.9
$\beta 2$	0.999
λ	0.1

Table 3.2: Hyperparameter Values

The learning rate was initially taken as $3 \times 10^{-}5$. The learning rate is divided by 10, whenever the total validation loss stops decreasing. The model was trained on NVIDIA 1080 Ti with 12 GB RAM and it takes around 6-7 hours to converge for the CelebA dataset and 2-3 hours to converge for the UTKFace Dataset.

Chapter 4

Experiments, Results and Observations

We have proposed an MTL algorithm which detects gender, age and race of a given facial image. We proposed a method to combine the gender, age, and race losses a "Weighted Loss". We also proposed a method to "Early freeze" the shared network when some of the tasks converge. We have used two datasets UTKFace and CelebA datasets to assess the performance of the proposed method. We have compared the MTL model with single task baselines. We have also compared the proposed MTL model with the state-of-the-art methods.

4.1 Datasets

UTKFace and CelebA datasets were used for the training and assessment of the proposed MTL model. Below is the brief description of both the datasets.

- **UTKFace Dataset**
  UTKFace Dataset is a face dataset with around 20,000 facial images. Each of these image is labelled with age, gender and race/ethnicity. This dataset covers a large age area from 0 to 116 years. This dataset is developed keeping in mind different conditions in which the facial image could be taken e.g., different illumination

conditions, occluded faces, different facial poses and expressions etc. Below is the age distribution plot for Male and Female Category.

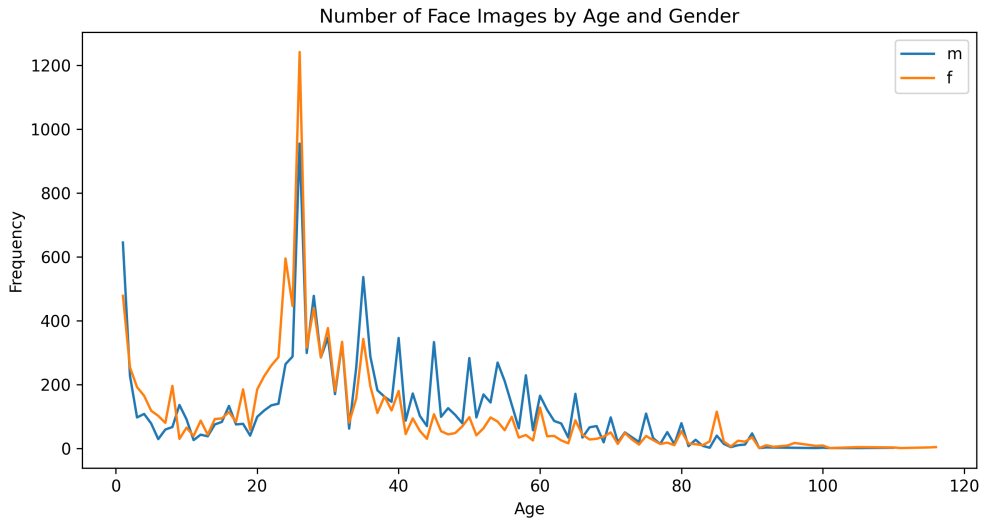


Figure 4.1: UTKFace Dataset Age Distribution per Gender Category (Male and Female)

The dataset has 5 race classes namely White, Black, Asian, Indian and Others. The data distribution of the race class for a given gender category is shown below. As we can see in Table 4.1 , there is a data imbalance among the race classes, and we need to counter this issue while training, otherwise more bias would be given to more dominant class (here White Class).

Race	**Total Persons**	**Male Count**	**Female Count**
White	10088	5477	4611
Black	4526	2318	2208
Asian	3434	1575	1859
Indian	3975	2261	1714
Others	1692	760	932

Table 4.1: UTKFace Dataset Distribution According to Race and Gender Classes

- **CelebA Dataset**
  CelebFaces Attributes Dataset (CelebA) has 202,599 celebrity facial images. Every

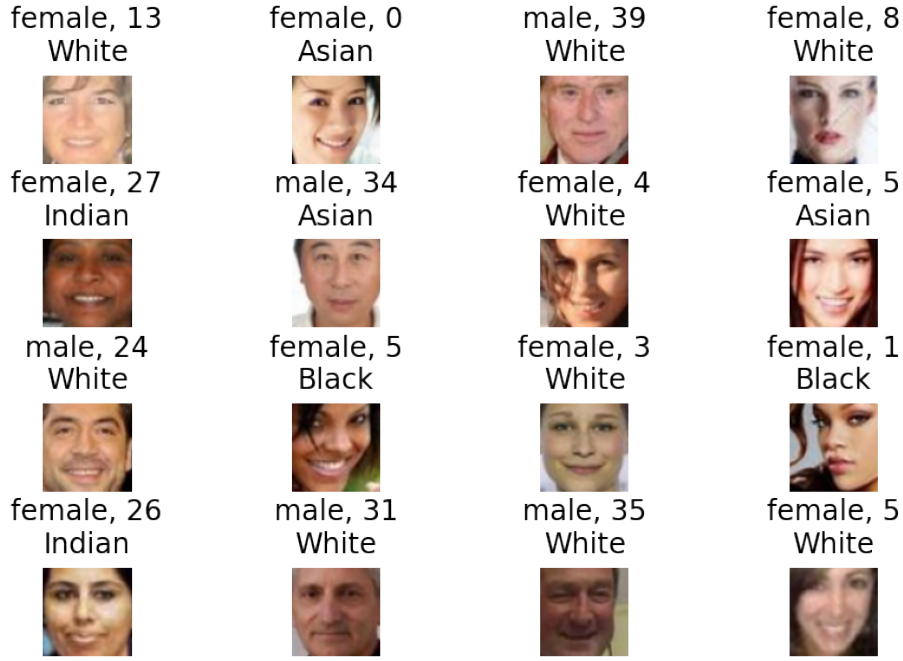


Figure 4.2: UTKFace Dataset Samples

image is labelled with 40 attributes e.g, Male, Young, Eyeglasses,Wearing Hat etc. The dataset has variations in background, pose and illumination and can be used for face detection, recognition, gender prediction (Male or Female), age classification (Young or Old) etc.

Since our work focuses on gender prediction and age estimation task, we have considered age and gender attributes only out of all 40 attributes in the CelebA Dataset. Following tables provide data distribution of Gender and Age Classes.

Task	**Male Count**	**Female Count**
Gender	68261	94509
Task	**Young Count**	**Old Count**
Age	126788	35982

Table 4.2: CelebA Dataset Distribution According to Gender and Age Classes

As we can see in Table 4.2 , there is a data imbalance in Gender class and Age

Class, and we need to counter this issue while training, otherwise more bias would be given to more dominant class (here Young Class and Female Class).

Figure 4.3: CelebA Dataset Samples

4.2 Dataset Experimental Protocol

- **UTKFace Dataset Protocol**
  We have followed protocol given in *"CORAL: Consistent Rank Logits for Ordinal Regression with Convolutional Neural Networks"* paper published by Raschka et al. [1].

Split	Number of examples
Train	10517
Validation	2630
Test	3287

Table 4.3: Dataset Split in UTKFace Dataset

Raschka et al. [1] have considered age ranging from 21 to 60 years and labeled them to the range of 0 to 39. They have provided CSV files for train and test splits. Train split is further divided into train and validation split in the ratio 80:20. Table 4.3 shows the number of samples present in each split.

- **CelebA Dataset Protocol**
  For CelebA dataset, we have followed protocol provided by the CelebA dataset. Table 4.4 shows the data split for CelebA Dataset.

Split	Number of examples
Train	162770
Validation	19867
Test	19963

Table 4.4: Dataset Split in CelebA Dataset

4.3 Pretrained Models for Facial Feature Vector Extraction

- As explained in Section 3.2, the cropped facial image is passed through the face-detection pretrained model. We removed the last layer of the pretrained model to get a feature vector for a given input facial image.
- The convolution filters of the pretrained model contain crucial information about the face. The lower layers detects the edges and shapes while the upper layers have filters to identify eyes, nose, skin colour, textures etc. [21]. Thus the feature vector is rich in facial information which is then fed into the MTL layers.
- In training phase, apart from the training of the MTL layers, the pretrained model is also fine-tuned on the dataset under consideration. This pretrained model thus act as a shared network among the different task (Figure 3.2).

- We have done experiments with 5 pretrained models. Brief details of these models are summarised in the table below.

Network Architecture	Training Dataset	Feature Dimension
LightCNN	MS-Celeb-1M	256
Sphereface	CASIA-WebFace	512
Inception-ResNet	VGGFace2	512
ResNet-50	MS-Celeb-1M and finetuned on VGGFace2 dataset	2048
SE-ResNet-50-128D	MS-Celeb-1M and finetuned on VGGFace2 dataset	128

Table 4.5: Pretrained Models Trained for Face Detection Task

4.4 Results and Observations

4.4.1 Comparison with Baselines on UTKFace Dataset

Tables 4.6 to 4.10 show the result for UTKFace Dataset when pretrained models used are LightCNN, Sphereface, Facenet, ResNet-50 and SE-ResNet-50-128D respectively.

- The "Method" Column in these tables represents the type of network which follows the pretrained model.
- The first row of these table with method name "Single Task baseline" shows the result when the output of the bottleneck layer (penultimate layer) of pretrained model is directly fed into an output layer. This output layer estimates the age in case of Age Estimation and in case of Gender and Race Classification, this output layer calculates the softmax of the output scores for different classes. Every task has a separate baseline model, thus each model is trained separately.
- The second row of table shows the result when pretrained model is followed by a single task neural network. Each of these are trained separately. Thus to get the results, we train each task in MTL model (Figure 3.2) separately and do not consider the "Loss Weight Layer".
- Third row of the table shows the result when pretrained model is followed by the MTL network (Refere to Figure 3.2 , Section 3.2).

- For gender and race classification, greater the Accuracy , better is the model. For age estimation, lower the Age Mean Absolute error (MAE), better is the model.
- Figures 4.4, 4.5, and 4.6 show the "Single Task Neural Network Baseline" for UTKFace Dataset. For CelebA dataset, we don't consider the Race baseline. Further, Age Task baseline has 128x2 as last layer since age task is binary classification task in CelebA dataset. For "Single Task Baseline", we only have an output layer followed by the pretrained model.

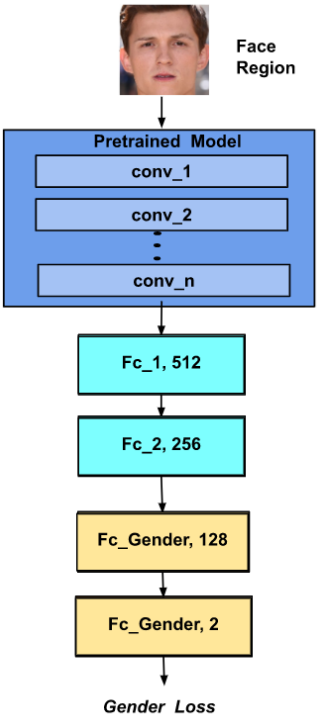


Figure 4.4: Gender Prediction Single Task Model

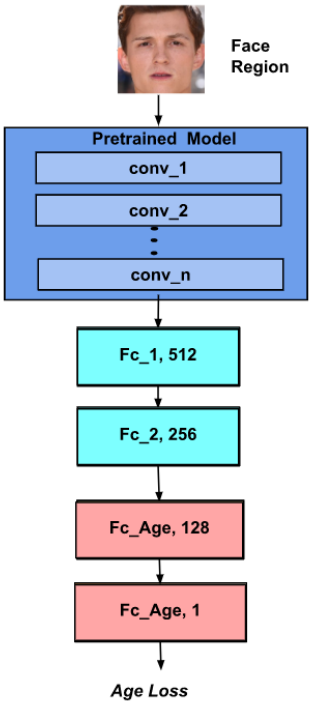


Figure 4.5: Age Prediction Single Task Model

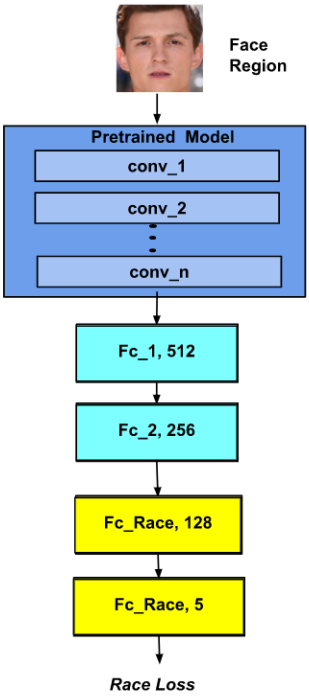


Figure 4.6: Race Prediction Single Task Model

- **In Tables 4.6 to 4.10, Gender and Race accuracy are written in percentage, while Age MAE is in years.**

Method (LightCNN)	Gender Accuracy	Race Accuracy	Age MAE
Single Task baseline	91.69	73.01	7.14
Single Task Neural Network	91.24	74.39	6.35
Proposed Multi-Task Learning Network	**96.56**	**84.76**	**4.82**

Table 4.6: Comparison of Methods when Pretrained Model is LightCNN on UTKFace

Method (Sphereface)	Gender Accuracy	Race Accuracy	Age MAE
Single Task baseline	90.75	67.52	8.69
Single Task Neural Network	91.36	69.15	6.61
Proposed Multi-Task Learning Network	**97.52**	**85.29**	**5.00**

Table 4.7: Comparison of Methods when Pretrained Model is Sphereface on UTKFace

Method (Inception-ResNet)	Gender Accuracy	Race Accuracy	Age MAE
Single Task baseline	87.01	77.20	8.93
Single Task Neural Network	88.03	80.10	7.39
Proposed Multi-Task Learning Network	**97.02**	**85.34**	**6.36**

Table 4.8: Comparison of Methods when Pretrained Model is Inception-ResNet (Facenet) on UTKFace

Method (ResNet-50)	Gender Accuracy	Race Accuracy	Age MAE
Single Task baseline	93.55	80.61	4.98
Single Task Neural Network	94.82	82.51	4.81
Proposed Multi-Task Learning Network	**98.41**	**86.87**	**4.72**

Table 4.9: Comparison of Methods when Pretrained Model is ResNet-50 on UTKFace

Method (SE-ResNet-50-128D)	Gender Accuracy	Race Accuracy	Age MAE
Single Task baseline	93.12	80.79	5.43
Single Task Neural Network	92.58	80.97	5.22
Proposed Multi-Task Learning Network	**98.47**	**87.56**	**4.68**

Table 4.10: Comparison of Methods when Pretrained Model is SE-ResNet-50-128D on UTK-Face

4.4.2 Comparison with Baselines on CelebA Dataset for Different Pretrained Models

Tables 4.11 to 4.15 show the result for CelebA Dataset when pretrained models used are LightCNN, Sphereface, Facenet, ResNet-50 and SE-ResNet-50-128D respectively. Since Race attribute is not provided in the CelebA dataset, so during training, we take the loss component from Race task as zero and only combine the age and gender loss using a 2x1 weight layer. For CelebA, both gender and age are classification tasks and Accuracy (in percentage) is the evaluation metric used to assess the performance of the MTL Network.

Method (LightCNN)	**Gender Accuracy**	**Age Accuracy**
Single Task baseline	84.86	76.02
Single Task Neural Network	90.68	77.54
Proposed Multi-Task Learning Network	**97.94**	**87.99**

Table 4.11: Comparison of Methods when Pretrained Model is LightCNN on CelebA Dataset

Method (Sphereface)	**Gender Accuracy**	**Age Accuracy**
Single Task baseline	86.03	73.95
Single Task Neural Network	91.33	82.31
Proposed Multi-Task Learning Network	**98.11**	**86.89**

Table 4.12: Comparison of Methods when Pretrained Model is Sphereface on CelebA Dataset

Method (Inception-ResNet)	**Gender Accuracy**	**Age Accuracy**
Single Task baseline	93.93	81.25
Single Task Neural Network	96.89	86.52
Proposed Multi-Task Learning Network	**98.59**	**87.62**

Table 4.13: Comparison of Methods when Pretrained Model is Inception-ResNet on CelebA

Method (ResNet-50)	Gender Accuracy	Age Accuracy
Single Task baseline	94.90	82.01
Single Task Neural Network	96.85	86.30
Proposed Multi-Task Learning Network	**98.79**	**86.83**

Table 4.14: Comparison of Methods when Pretrained Model is ResNet-50 on CelebA

Method (SE-ResNet-50-128D)	Gender Accuracy	Age Accuracy
Single Task baseline	92.43	78.92
Single Task Neural Network	97.23	84.75
Proposed Multi-Task Learning Network	**98.82**	**87.53**

Table 4.15: Comparison of Methods when Pretrained Model is SE-ResNet-50-128D on CelebA

From the results on CelebA dataset and UTKFace dataset, we got best result with SE-ResNet-50-128D pretrained model for both of the datasets. We will use this result to compare with the recent state-of-the-art methods. We will identify this architecture as "**MTL+SEResnet** model" in later sections.

4.4.3 Comparison of Gender Accuracy on CelebA Dataset

- Figure 4.7 shows a bar-graph which compares the proposed MTL method with previous works on CelebA Gender Prediction Task. It shows that our proposed model gives best performance on the gender recognition task. It only ties with All-in-one CNN [15] and outperforms the other methods.
- We have compared our method with FaceTracer [6], PANDA-W [22], R-CNN_Gender [14], Walk and Learn [19], PANDA-1 [22], Multitask_Face [14], HyperFace [14], LNets + ANet [8], HF-ResNet [14], DMM-CNN [12], and All-in-one CNN [15].
- Hyperface and HF-ResNet [14] also used Multi-Task method with their proposed MTL architecture but our model outperforms this method in gender classification.
- In Hyperface [14], authors mentioned that they combined the task-specific loss by a weighted sum of losses and the weights were given empirically. Giving weights to losses in such a way can give a higher bias to one loss than others.

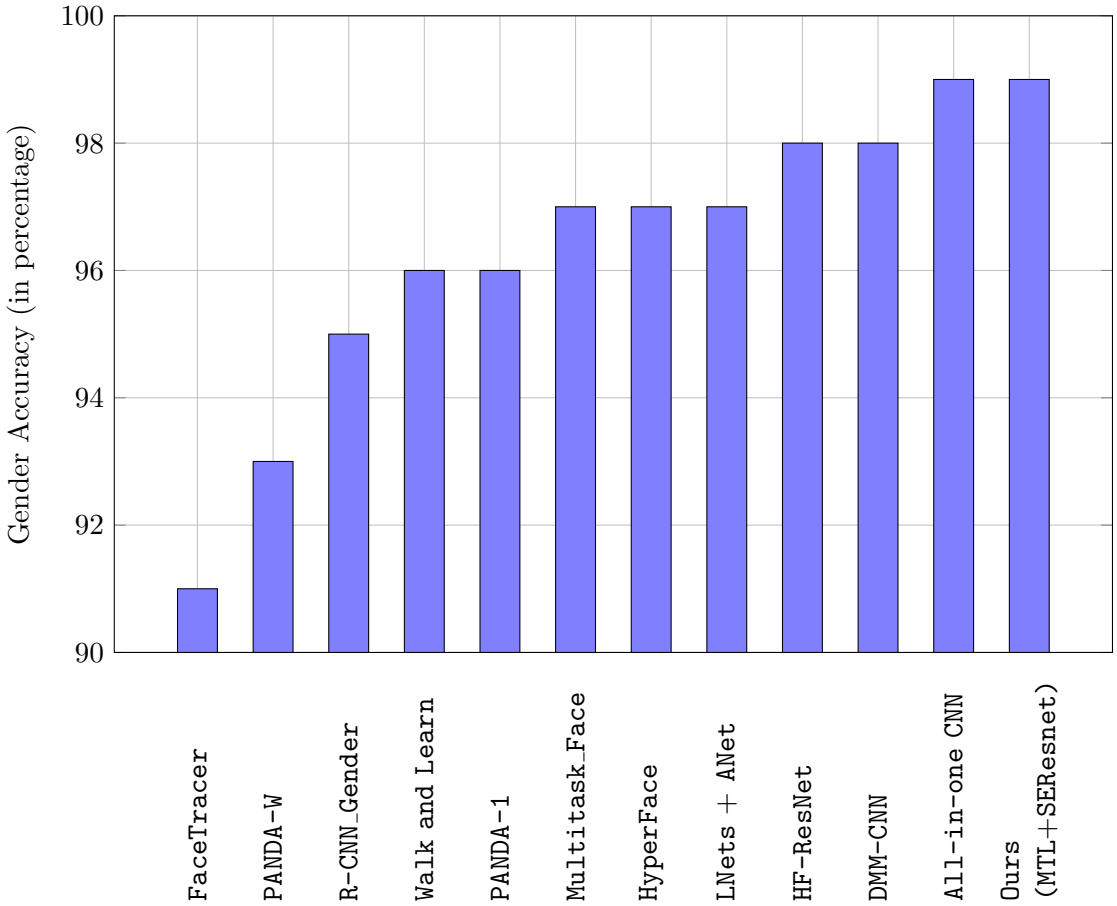


Figure 4.7: Comparison of Gender Recognition Accuracy (in Percentage) on CelebA Dataset

In our research, we proposed a regularised weighted loss function (Section 3.4) which learns the weights to be given to the losses during training phase. The model thus learns the optimal weights that are to be given to the losses, so that the performance of the model increase. Refer to Section 5.1 for analysis of performance improvement when proposed weighted loss is used.

- Further we also used the proposed Early Freeze method (Section 3.5) while training, which helped the tasks to converge better in the multi-task learning environment. Refer to Section 5.2 for analysis of performance improvement when proposed Early freeze method is used.

4.4.4 Comparison of Age Prediction Accuracy on CelebA Dataset

- We have compared our method with PANDA-W [22], FaceTracer [6], PANDA-1 [22], Walk and Learn [19], LNets + ANet [8], and DMM-CNN [12].
- From Figure 4.8, we can see that our model outperforms the previous age prediction tasks except DMM-CNN [12].

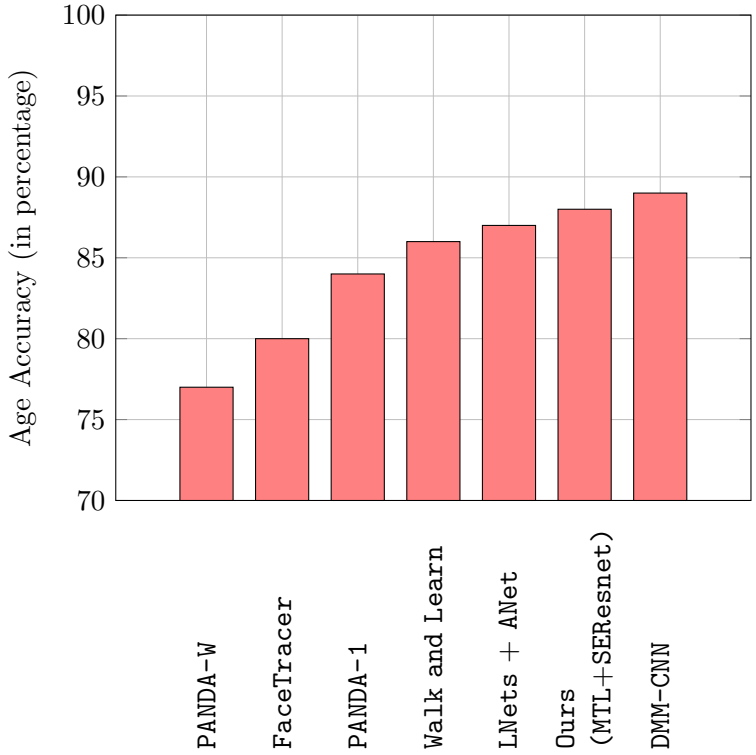


Figure 4.8: Comparison of Age Prediction Accuracy (in Percentage) on CelebA

- Since our objective was to predict the age, and gender from the facial image, we only took these attributes while training. On the other hand, objective of DMM-CNN method was to achieve best average accuracy over all the 40 attributes of CelebA dataset, so they took into consideration all the attributes while training. So it might be possible that due to more tasks involved, their model generalised better for Age prediction task.
- **On Gender prediction task, our method (98.82 %) outperforms DMM-CNN (98.29 %), while for Age prediction task, DMM-CNN (88.94 %)**

performs better than our method (87.53 %).

4.4.5 Comparison of Age Mean Absolute Error (MAE) on UTKFace Dataset

Method	Age MAE (in years)
CE-CNN	6.02
OR-CNN	5.60
CORAL-CNN	5.25
Ours (MTL+SEResnet)	**4.68**

Table 4.16: Comparison of MAE (in years) for Age Estimation Task on UTKFace Dataset

- We have used protocol provided by CORAL-CNN [1] for division of UTKFace Dataset into train, validation and test sets. The UTKFace dataset has age ranging from 0-116 years, but protocol provided by CORAL-CNN takes into consideration the age ranging from 21 to 60 years.
- OR-CNN and CORAL-CNN gave better result than the standard CE-CNN (Cross Entropy Loss). Our model performs best when pretrained model used is SE-Resnet-50-128D with age MAE of 4.68 years. Our model outperforms the state-of-the-art CORAL-CNN model by a difference of 0.57 years.
- The increase in the performance of our proposed work is due to the fact that our model learns simultaneously age, race and gender, thus is expected to have improved generalisation performance. On the other hand CORAL-CNN just utilised single task of age prediction and not using other attributes for learning.

4.4.6 Misclassified Samples from MTL+SEResnet on CelebA Dataset

Figure 4.9: Misclassified Images in Age Prediction Using MTL+SEResnet Model

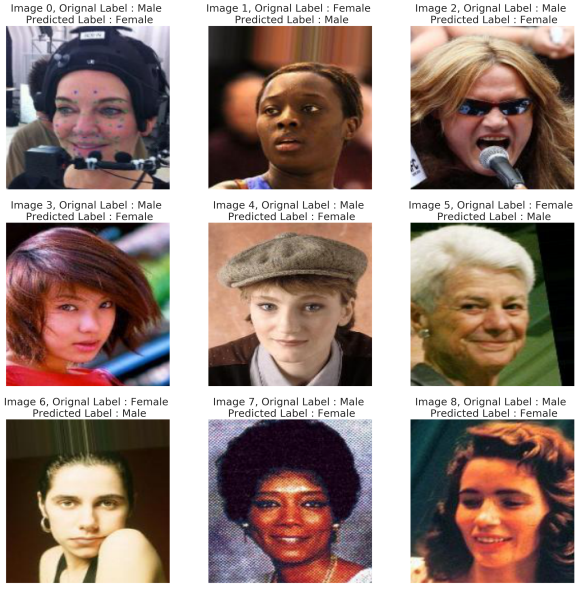


Figure 4.10: Misclassified Images in Gender Prediction Using MTL+SEResnet Model

Figures 4.9 and 4.10 show sample of images that were misclassified by MTL+SEResnet model.

4.4.7 Results on Real-World Images

We have implemented Python based application which takes a video file or camera as an input and simultaneously find age, race and gender attributes of each face detected in the frame. It first uses dlib-ml libray [5] to extract the facial regions in image. We crop out the facial regions and give it to the MTL network as batch of images. This model then simultaneously predict the gender, age and race attributes of the given facial images. Figure 4.11 shows the framework for testing real world image. Figures 4.12 and 4.13 shows the results of the proposed model on CelebA and UTKFace datasets respectively.

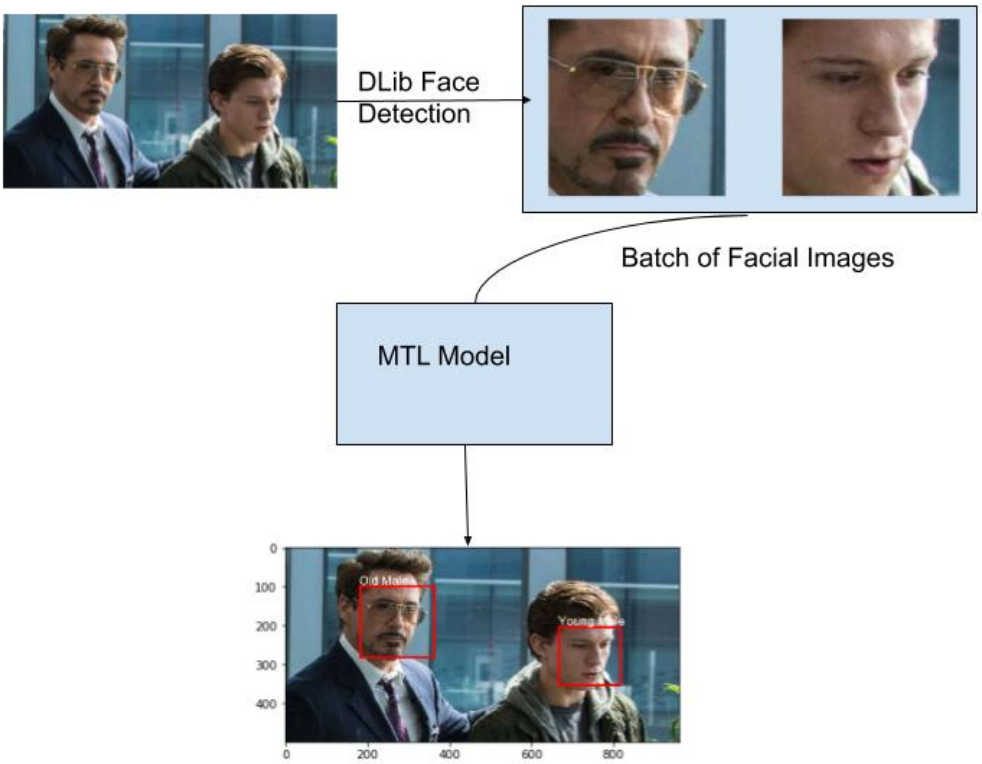


Figure 4.11: Testing Framework for Real-World Images

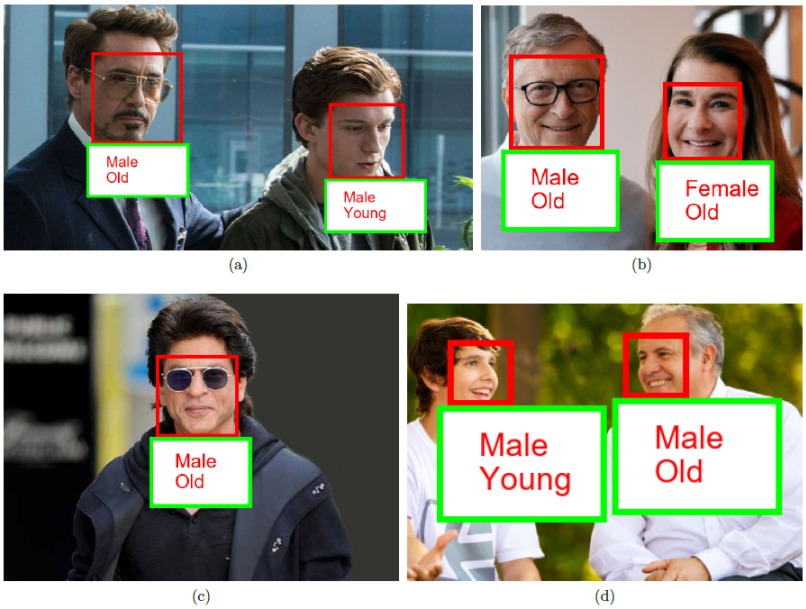


Figure 4.12: Simultaneous Face Attribute Detection from Real-World Images using MTL Model Trained on CelebA Dataset

Figure 4.13: Simultaneous Face Attribute Detection from Real-World Images using MTL Model Trained on UTKFace Dataset

Chapter 5

Analysis of Proposed MTL Method

In this section, we will analyse the improvement in performance when proposed weighted loss function and proposed early freezing are used.

5.1 Performance Improvement with Proposed Weighted Loss

5.1.1 CelebA Dataset

We carried out experiment to assess the effects of proposed weighted loss function on MTL+SERenet model and MTL+Resnet-50 model trained on CelebA dataset, Tables below show the results of the experiment.

Loss function	Gender Accuracy	Age Accuracy
Average Loss	98.74	87.18
Proposed Weighted Loss	98.82	87.53

Table 5.1: Table Showing Improvement in CelebA Results when Proposed Weighted Loss was Used Instead of Average Loss and Pretrained Model Used is SE-ResNet-50

From Table 5.1 and 5.2, we can see that using the proposed weighted loss function, we are getting better results as compared to average loss. Since in case of CelebA, both gender and age are binary classification tasks, their losses are somewhat similar initially, so we are not able to see high performance improvement. In a case where

Loss function	Gender Accuracy	Age Accuracy
Average Loss	98.72	86.61
Proposed weighted Loss	98.79	86.81

Table 5.2: Table Showing Improvement in CelebA Results when Proposed Weighted Loss was Used Instead of Average Loss and Pretrained Model Used is ResNet-50

there is a high difference between the loss magnitudes, our proposed loss function will show significant impact (Section 5.1.2).

5.1.2 UTKFace Dataset

We carried out experiment to assess the effect of the proposed weighted loss function on the MTL + Resnet-50 model trained on UTKFace dataset, Table 5.4 shows the results of the experiment.

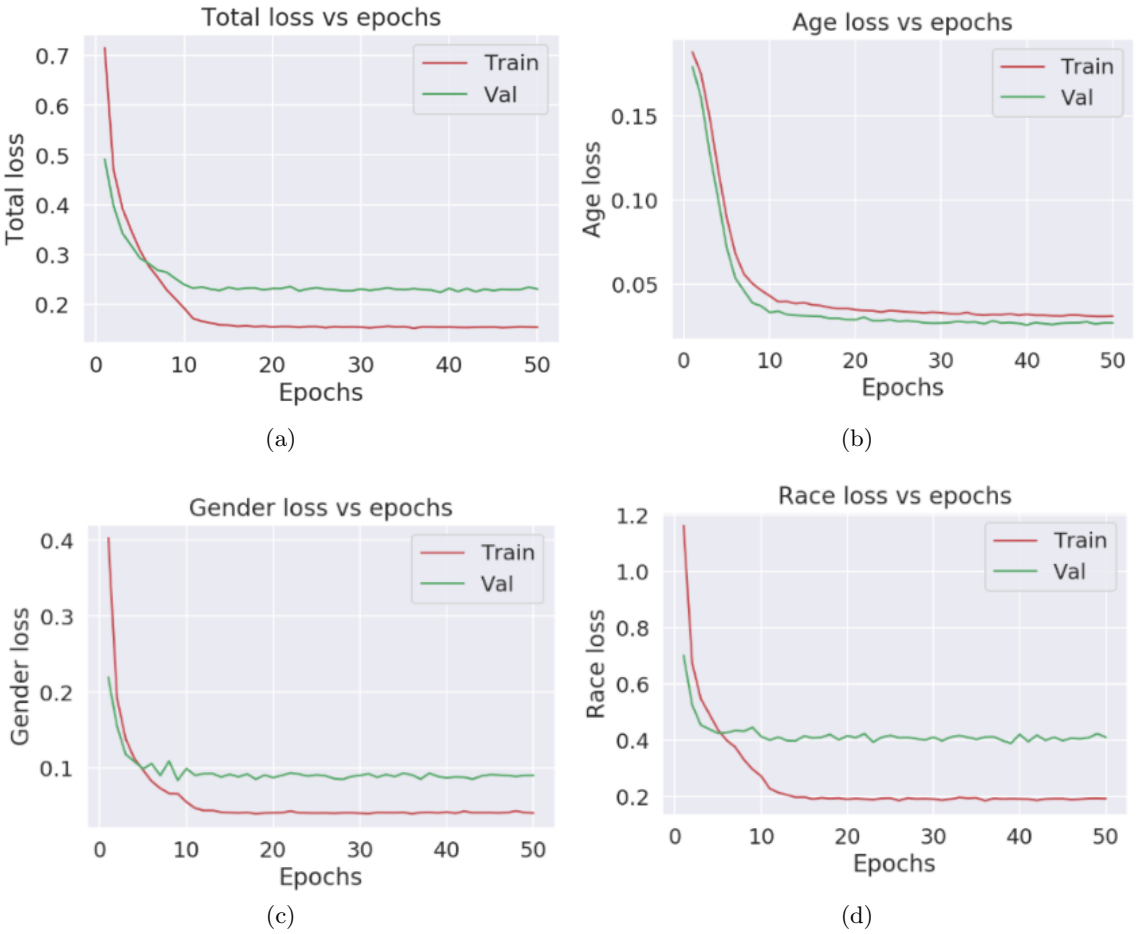
Loss function	Gender Accuracy	Race Accuracy	Age MAE
Average Loss	98.01	87.56	5.17
Proposed weighted Loss	98.47	87.56	4.68

Table 5.3: Table Showing Improvement in UTKFace Results when Proposed Weighted Loss was Used Instead of Average Loss and Pretrained Model Used is SE-ResNet-50

- From above table, we can see that when the proposed weighted loss is used instead of average loss, we get better results. The race loss was larger in magnitude than other losses (around double the gender loss at the start of the training). So race loss got more bias towards convergence. Also the age loss was even lesser than gender loss initially thus it got least bias, thus did not converge better.
- While race loss comes out to be almost same in both the loss functions, age and gender performance are reduced when average loss is used as the training was biased towards the race task due to large magnitude of race loss as compared to other losses.
- Thus, by using the the proposed weighted loss function, our model does not get biased towards any of the task loss, rather it learns the optimal weights to be given to the losses so that the model generalises better.

5.2 Performance Improvement with Proposed Early-Freeze Method

In this section, We have used MTL model with (SE-Resnet-50-128D as pretrained model) to assess the performance when the proposed Early Freeze method is used. Below are the loss versus iteration curves when Early Freeze algorithm is used (Figure 5.1) and when Early Freeze algorithm is not used (Figure 5.2).



(a) (b) (c) (d)

Figure 5.1: Loss vs Iterations Graphs when Early Freeze Method used

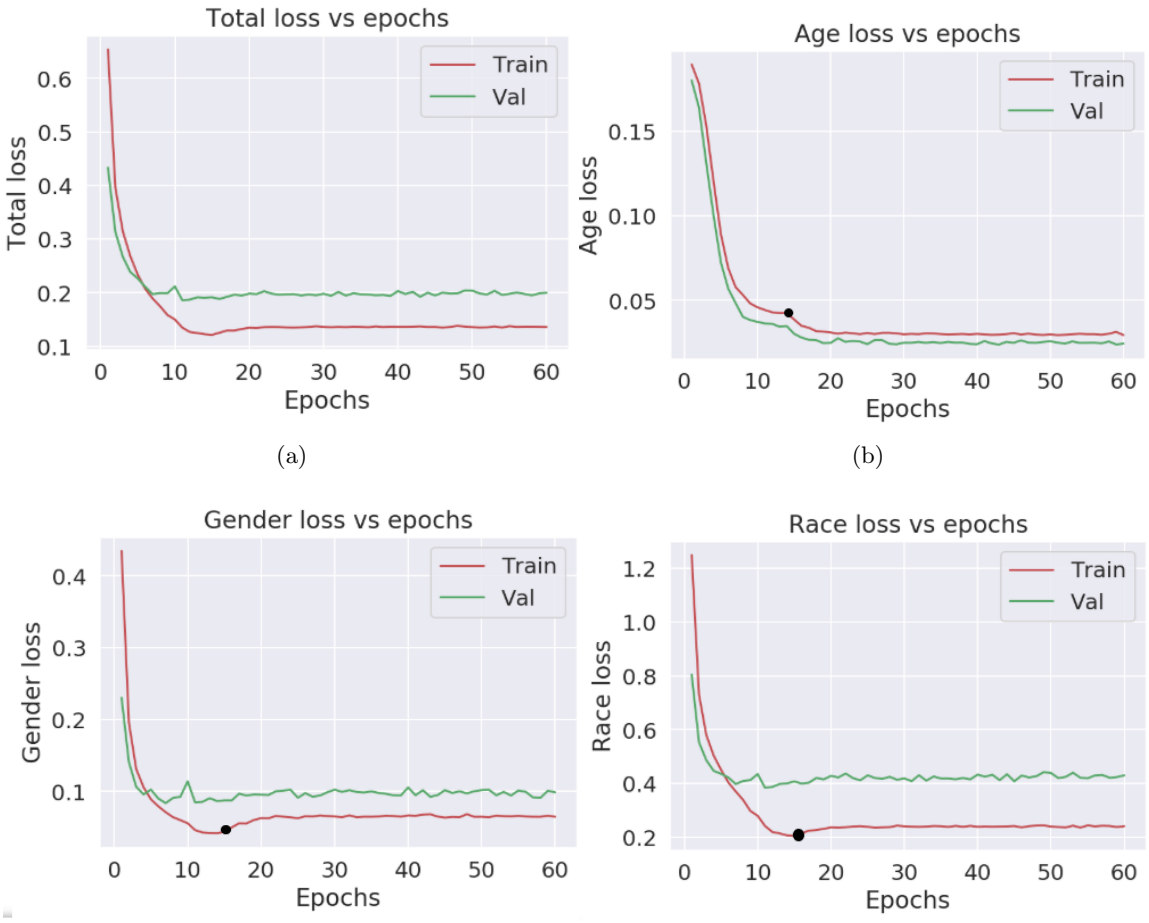


Figure 5.2: Loss vs Iterations Graphs when Early Freeze Method is not used

From the above figures, we can observe that when the proposed Early Freeze method is not used, the gender and race tasks have converged at epoch number 14, and after convergence, when they are stopped to learn via "Early Stop", there is increase in their losses due to the change in shared model parameters by non converged task (here age task). This can be seen by sudden increase in losses of converged tasks and decrease in loss of non-converged task (age task) after epoch number 14 (Shown by Black Dot in Figure 5.2). We found that the validation accuracy of gender and race at their convergence has got decreased when age task is converged at around 60 epochs. But this decrease in validation accuracy of already converged tasks does not happen when proposed Early Freeze method is used.

Loss function	Gender Accuracy	Race Accuracy	Age MAE
Early stopping of converged task	98.31	86.58	4.64
Proposed Early Freeze Method	98.47	87.56	4.68

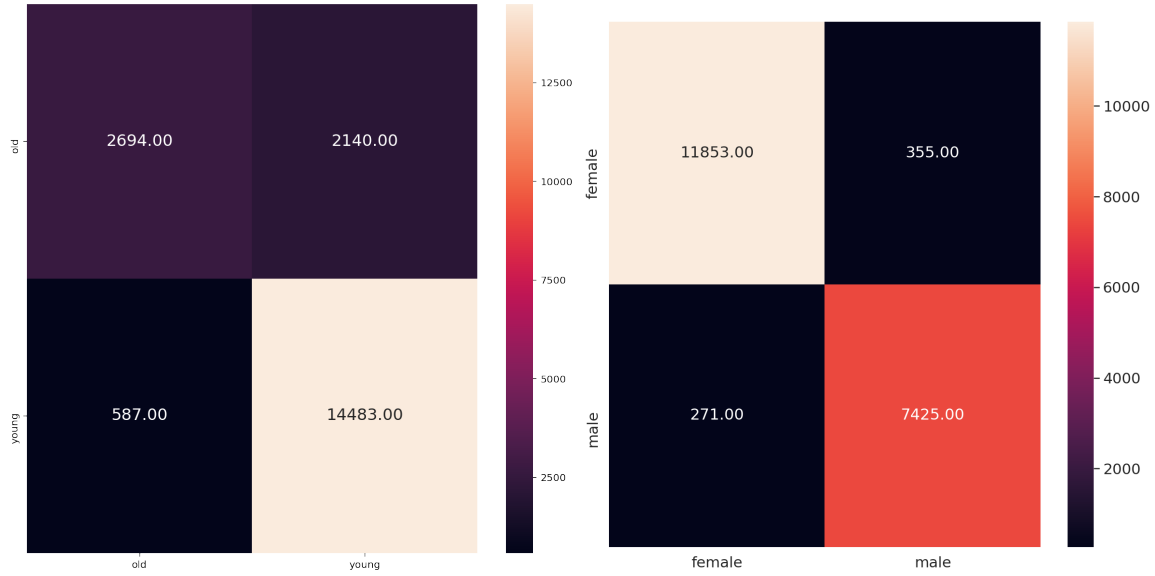
Table 5.4: Table Showing Improvement in UTKFace Results when Proposed Early Freezing was Used and Pretrained Model Used is SE-ResNet-50

From above table, we can see that the accuracy of the already converged tasks (race and gender) got decreased due to non converged tasks. This though, improves the Age MAE, but on the cost of decreasing accuracy of gender and race tasks. On the other hand, when using Early Freeze, performance of the converged tasks is not decreased as the shared layers are freezed and the non converged tasks still have scope to learn using its task-specific layers.

5.3 Mitigation of Gender and Age Bias due to Multi-Task Learning on CelebA Dataset

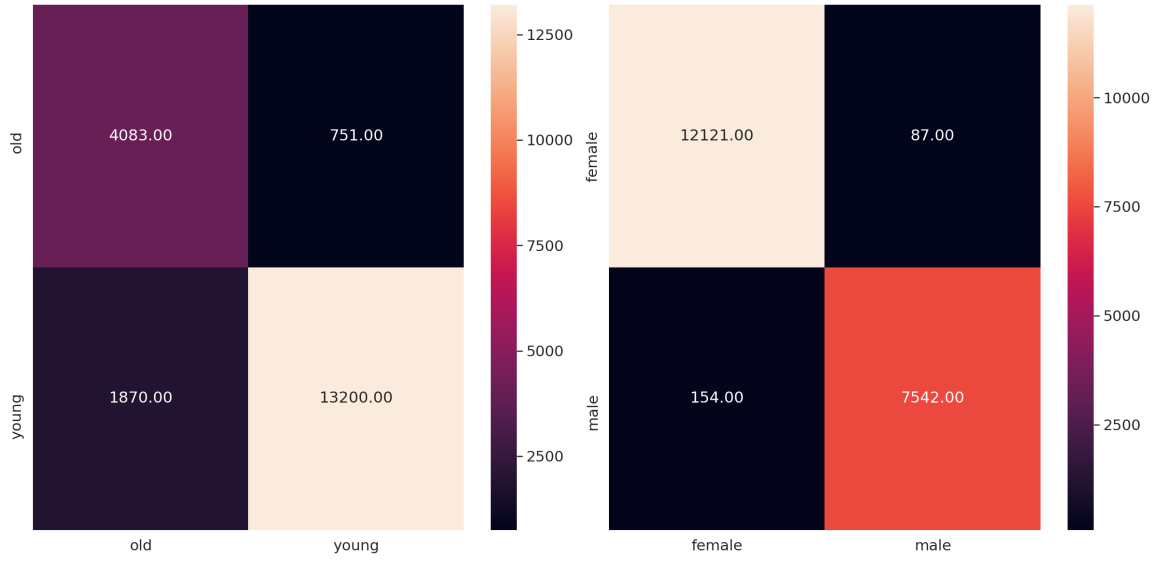
In this section, we compare the results of fully connected single task network and multi-task network on CelebA dataset with MTL + SEResnet as pretrained model to see if MTL helps in mitigating/ reducing the bias. As explained in 4.1, CelebA dataset has Young Count as 126788 and old count as 35982, and 68261 Male count and 94509 Female count. So there is class imbalance that can lead to giving bias to larger class while training.

When a single task neural network is used, Age accuracy is 86.30 % but class-wise accuracy shows that the model was biased towards the majority class (Young) giving 96.10 % accuracy for Young class and only 55.73 % accuracy to Old Class. When MTL is used, Age accuracy is 86.83 % and class-wise age accuracy suggests young class has 87.59 % and Old class has 84.46 % age accuracy. Thus we can see that the biasing in case of single task model towards young class is decreased when MTL model is used and both Young and Old classes have similar performance.



(a) Age Confusion Matrix (b) Gender Confusion Matrix

Figure 5.3: Confusion Matrix for Single Task Neural Network Using ResNet-50 Pretrained Model on CelebA Dataset. Confusion Matrix Clearly Depicts Bias Given to Young Task than the Old Task



(a) Age Confusion Matrix (b) Gender Confusion Matrix

Figure 5.4: Confusion Matrix for Proposed MTL Network Task Using ResNet-50 Pretrained Model on CelebA Dataset

5.4 Hyperparameter Tuning for Regularisation Hyperparameter (λ) in the Proposed Weight Loss

We have tuned values of λ given in equaltion 3.5 with values 0.001,0.01,0.1,1. We got best results with $\lambda = 0.1$ for any of the pretrained model. Table 5.5 shows results for different values of λ for SE-ResNet-50 pretrained model.

λ	**Gender Accuracy**	**Race Accuracy**	**Age MAE**
0.001	98.31	86.83	4.76
0.01	98.39	86.97	4.74
0.1	98.47	87.56	4.68
1	98.23	86.79	4.82

Table 5.5: Table Showing Results for Different Values of Hyperparameter λ and Pretrained Model Used is SE-ResNet-50

Chapter 6

Conclusions and Future Work

In this research, we proposed a multi-task architecture which simultaneously detects age, gender and race when UTKFace dataset is used for training and, age and gender when CelebA dataset is used for training. We have compared our method with single-task learning baselines. Our method gives better performance as compared to the simple baselines and single task neural networks. Improved results using MTL indicates that the model is able to learn more effectively when age, race and gender tasks were learned simultaneously by leveraging the similarity among the tasks. Our model gives best results when SE-ResNet-50-128D is used as a pretrained model, for both the datasets. We proposed a regularised weighted loss function in which instead of giving weights to the losses empirically, the model itself learns the best weights that should be given to each of the loss. This helps to avoid the situation in which the model might prefer one task over others. We also proposed Early Freeze method in which we freeze the shared layers as soon as some task gets converged. We compared the best model with previous works in CelebA and UTKFace dataset, and it outperforms most of the state-of-the-art models.

In future, we plan to extend the work by assessing the performance of the proposed MTL architecture using more datasets and include face detection, localisation and pose estimation task along with age, gender and race recognition.

Bibliography

[1] Cao, W., Mirjalili, V., and Raschka, S. Rank-consistent Ordinal Regression for Neural Networks. *arXiv preprint arXiv:1901.07884* (2019).

[2] Caruana, R. Multitask Learning: A Knowledge-Based Source of Inductive Bias. *ICML* (1993).

[3] Caruana, R. Algorithms and Applications for Multitask Learning. In *ICML* (1996), pp. 87–95.

[4] Ehrlich, M., Shields, T. J., Almaev, T., and Amer, M. R. Facial Attributes Classification using Multi-Task Representation Learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (2016), pp. 47–55.

[5] King, D. E. Dlib-ml: A machine learning toolkit. *The Journal of Machine Learning Research 10* (2009), 1755–1758.

[6] Kumar, N., Belhumeur, P., and Nayar, S. Facetracer: A Search Engine for Large Collections of Images with Faces. In *European conference on computer vision* (2008), Springer, pp. 340–353.

[7] Levi, G., and Hassner, T. Age and Gender Classification using Convolutional Neural Networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (2015), pp. 34–42.

[8] Liu, Z., Luo, P., Wang, X., and Tang, X. Deep Learning Face Attributes in the Wild. In *Proceedings of the IEEE international conference on computer vision* (2015), pp. 3730–3738.

[9] Liu, Z., Luo, P., Wang, X., and Tang, X. Large-scale Celebfaces Attributes (celeba) Dataset. *Retrieved August 15* (2018). URL `http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html`.

[10] Long, M., Cao, Z., Wang, J., and Philip, S. Y. Learning Multiple Tasks with Multilinear Relationship Networks. In *Advances in neural information processing systems* (2017), pp. 1594–1603.

[11] Lu, Y., Kumar, A., Zhai, S., Cheng, Y., Javidi, T., and Feris, R. Fully-Adaptive Feature Sharing in Multi-Task Networks with Applications in Person Attribute Classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 5334–5343.

[12] Mao, L., Yan, Y., Xue, J.-H., and Wang, H. Deep Multi-task Multi-label CNN for Effective Facial Attribute Classification. *IEEE Transactions on Affective Computing* (2020). doi: `https://doi.org/10.1109/TAFFC.2020.2969189`.

[13] Misra, I., Shrivastava, A., Gupta, A., and Hebert, M. Cross-Stitch Networks for Multi-Task Learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 3994–4003.

[14] Ranjan, R., Patel, V. M., and Chellappa, R. Hyperface: A Deep Multi-Task Learning Framework for Face Detection, Landmark Localization, Pose Estimation, and Gender Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence 41*, 1 (2017), 121–135.

[15] Ranjan, R., Sankaranarayanan, S., Castillo, C. D., and Chellappa, R. An All-In-One Convolutional Neural Network for Face Analysis. In *12th IEEE International Conference on Automatic Face & Gesture Recognition* (2017), IEEE, pp. 17–24.

[16] Ruder, S. An Overview of Multi-Task Learning in Deep Neural Networks. *arXiv preprint arXiv:1706.05098* (2017).

[17] Ruder, S., Bingel, J., Augenstein, I., and Søgaard, A. Latent Multi-Task Architecture Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence* (2019), vol. 33, pp. 4822–4829.

[18] Wang, F., Han, H., Shan, S., and Chen, X. Deep Multi-Task Learning for Joint Prediction of Heterogeneous Face Attributes. In *IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)* (2017), IEEE, pp. 173–179.

[19] Wang, J., Cheng, Y., and Schmidt Feris, R. Walk and Learn: Facial Attribute Representation Learning from Egocentric Video and Contextual Data. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 2295–2304.

[20] Yi, D., Lei, Z., and Li, S. Z. Age Estimation by Multi-scale Convolutional Network. In *Asian conference on computer vision* (2014), Springer, pp. 144–158.

[21] Zeiler, M. D., and Fergus, R. Visualizing and Understanding Convolutional Networks. In *European conference on computer vision* (2014), Springer, pp. 818–833.

[22] Zhang, N., Paluri, M., Ranzato, M., Darrell, T., and Bourdev, L. Panda: Pose Aligned Networks for Deep Attribute Modeling. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2014), pp. 1637–1644.

[23] Zhang, Zhifei, S. Y., and Qi, H. Age Progression/Regression by Conditional Adversarial Autoencoder. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. URL `https://susanqq.github.io/UTKFace/`.

[24] Zhang, Y., and Yang, Q. A Survey on Multi-task Learning. *arXiv preprint arXiv:1707.08114* (2017).