Age Estimation on images of the face based on CNN

Ke LIANG

May 5, 2020

Contents

1 Abstract

Nowadays, the age estimation is an interesting and important task in the area of pattern recognition and machine learning. The work of mine is to estimate the age of the human based on the face images by using Deep learning to build models. The dataset is the APPA-REAL dataset with the 224x224 image labeled with apparent age and real age. I regarded the VGG-16 model as the baseline model, which was used by the state of art method, named "DeX Expectation"[1]. I also tested se_resnet50 model regarding to the paper[2] [3]. Besides, I have built five CNN models for the problem together with the data augmentation technique, transfer learning technique and down-sampling technique. In the end, the Modified Resnet + Transfer learning model outperformed other models of mine.

2 Introduction

Nowadays, the age estimation is an important and interesting task in the area of computer vision, pattern recognition and machine learning. This problem can be treated as either a regression or classification problem. There are lots of method for this problem from the traditional machine learning method, which is using feature extraction and selection method, to deep learning method, which is using neural network. The state of art method of this problem is the Deep Expectation (DEX) Method used in the paper "Deep expectation of real and apparent age from a single image without facial landmarks" [1].

The motivation for me to do this can be concluded into 3 aspects. First, from the course point of the view, it is can be treated both regression and classification problems which are related to our course. Second, it is executable due to lots of research on this topic. Third, for myself, I am really interested in this topic which is related to our daily life.

In my term project, I estimated the age of the human based on the face images by using Deep learning to build models. The dataset that I used is APPA-REAL dataset including the face images with apparent age labels and real age labels. Based on this dataset, I will build my own CNN model to do estimation both on apparent age and real age. I will treat this problem as the regression problems. It is unstable if directly training a CNN model for regression problem. Based on this, what I have done is to first train a CNN model for classification problem, then compute the expect value of the output of the softmax-normalized layer as the predicted age. I regard the VGG-16 model used in the Dex Method in paper [1] and the se_resnet50 model used in paper [2] and network [3] as the baseline models.

I use python language to implement my idea together with the Pytorch library. I built 5 models in my project, where the data augmentation technique, transfer learning technique and downsampling technique are also introduced. The evalutaion metric of the project are Error and Mean Absolute Error(MAE). The reason why I do not use accuracy as the metric is that even human can not guess the age of a person just based on the face information. Then the accuracy will be continuous low for most of the model. This is explained more in the method chapter.

In the end, the Modified Resnet + Transfer model is the greatest model of mine. While it is good for apparent age estimation, instead of the real age estimation. More details will be explained in the results and discussion chapters.

2.1 Goals

In my term project, I treat the "age estimation" problem as a Regression problem. My goal is to estimate the age of the human based on the face images by using Deep learning to build models. There are two parts for estimation, including apparent age estimation and real age estimation.

3 Related Work

The state of the art I found for this topic was the work finished by three scientist named Rasmus Rothe, Radu Timofte, Luc Van Gool, and their work was shown in the paper "Deep expectation of real and apparent age from a single image without facial landmarks" [1] [4].

3.1 The dataset for the related Work

In the paper, it used 5 different datasets for both real and apparent age, including IMDB−WIKI dataset, FG-NET dataset, MORPH dataset, CACD dataset and LAP dataset.

The IMDB−WIKI dataset is built with the images of celebrities from IMDb and Wikipeida including 100000 most popular actors together with their birth, name, gender and all the images related to the person and also crawling all the profile images and information from pages of people from Wikipedia. In the IMDB−WIKI dataset, it contains 523051 face images: 460723 face images including 20284 people from IMDb and 62328 people from Wikipedia, and each celebrity has about 23 images on average[1].

FG−NET dataset is short for "The Face and Gesture Recognition Research Network (FG−NET)" with 1002 images and on average there are 12 images for each 82 subjects of the age ranging from 0 to 69[1].

MORPH dataset is short for "The Craniofacial Longitudinal Morphological Face Database ". And this dataset contains more than 55000 shots with the age ranging from 16 to 77[1].

CACD is short for "The Cross-Age Celebrity Dataset (CACD)". This dataset contains 163446 images of 2000 celebrities[1] and it is split into 3 parts for the training, validation and testing set with the ratio 1800:80:120.

LAP is the Chalearn LAP dataset. It contains 4699 images[1] and it is split into 3 parts for the training, validation and testing set with the ratio 2476:1136:1087.

3.2 Related work on the Pattern Recognition Approach

The method of the state of the art of the estimation of the ages for images is Deep EXpectation method. The method is based on convolutional neural networks (CNNs) which is VGG−16 architecture and they pretrained their model on ImageNet first for image classification. The method they proposed contains the following parts:(1)Face alignment, which is finishing the face detection on the input images and cropping the face they need for next step, can be regarded as a prepossess (2) Using CNN with VGG−16 architecture to doing age estimation, and using softmax layer after the architecture to get the expected value E with the 101 dimensional output layer regarding to the ages from 0 to 95 [1] [4]. The following figure 1 is from the paper to show the flowchart of the method they proposed.

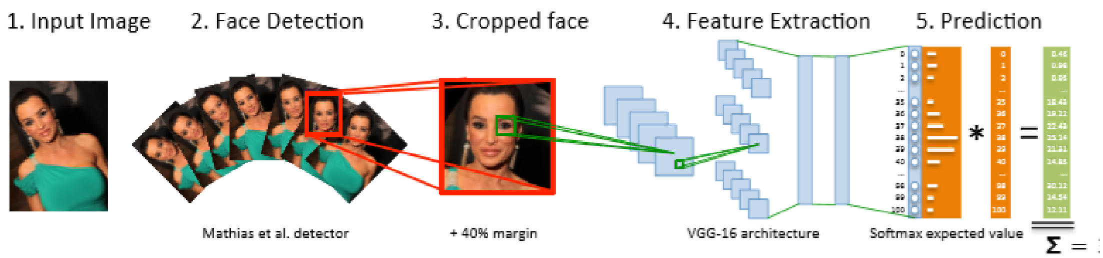
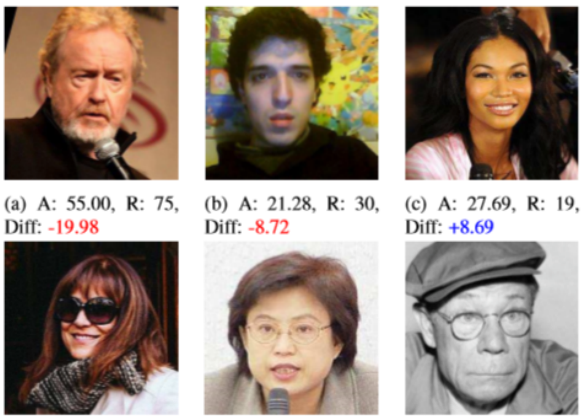


Figure 1: The pipeline of the state of the art method[1]

4 Data

4.1 Dataset Description

I used APPA−REAL dataset[5] [6] to finish the term-project. The APPA−REAL database contains 7,613 images with associated real and apparent age labels. The total number of apparent votes is around 250,000, and here are figures in the dataset shown below (figure 2).



(a) A: 55.00, R: 75, Diff: -19.98
(b) A: 21.28, R: 30, Diff: -8.72
(c) A: 27.69, R: 19, Diff: +8.69
(d) A: 37.46, R: 53, Diff: -15.60
(e) A: 44.28, R: 32, Diff: +18.28
(f) A: 71.00, R: 55, Diff: +16.00

Figure 2: Examples of real-apparent age biases on APPA−REAL dataset. Apparent (A), real age (R), and difference A−R (Diff) are shown for each face image[3]

On average I have around 38 votes per each image and this makes the average apparent age very stable (0.3 standard error of the mean)[3].

The dimension of the input images is 224x224, and the comparison of the dataset are shown in the figure 3.

Database	# of faces	# of subjects	Age range	Age type	Environment
FG-NET	1,002	82	0-69	Real Age	Uncontrolled
GROUPS	28,231	28,231	0-66+	Age Group	Uncontrolled
PAL	580	580	19-93	Age Group	Uncontrolled
FRGC	44,278	568	18-70	Real Age	Partly Controlled
MORPH2	55,134	13,618	16-77	Real Age	Controlled
YGA	8,000	1,600	0-93	Real Age	Uncontrolled
FERET	14,126	1,199	-	Real Age	Partly Controlled
Iranian face	3,600	616	2-85	Real Age	Uncontrolled
PIE	41,638	68	-	Real Age	Controlled
WIT-BD	26,222	5,500	3-85	Age Group	Uncontrolled
Caucasian Face Database	147	-	20-62	Real Age	Controlled
LHI	8,000	8,000	9-89	Real Age	Controlled
HOIP	306,600	300	15-64	Age Group	Controlled
Nis Web-Collected Database	219,892	-	1-80	Real Age	Uncontrolled
OUI-Adience	26,580	2,284	0-60+	Age Group	Uncontrolled
IMDBWIKI	523,051	20,284+	0-100	Real Age	Uncontrolled
APPA-REAL	7,591	7,000+	0-95	Real and Apparent Age	Uncontrolled

Figure 3: Comparison of the dataset[3]

In this table, it is easy to notice that the dataset APPA−REAL has great range of the age and quantities of the image. The distribution of the ages in the dataset is shown below.

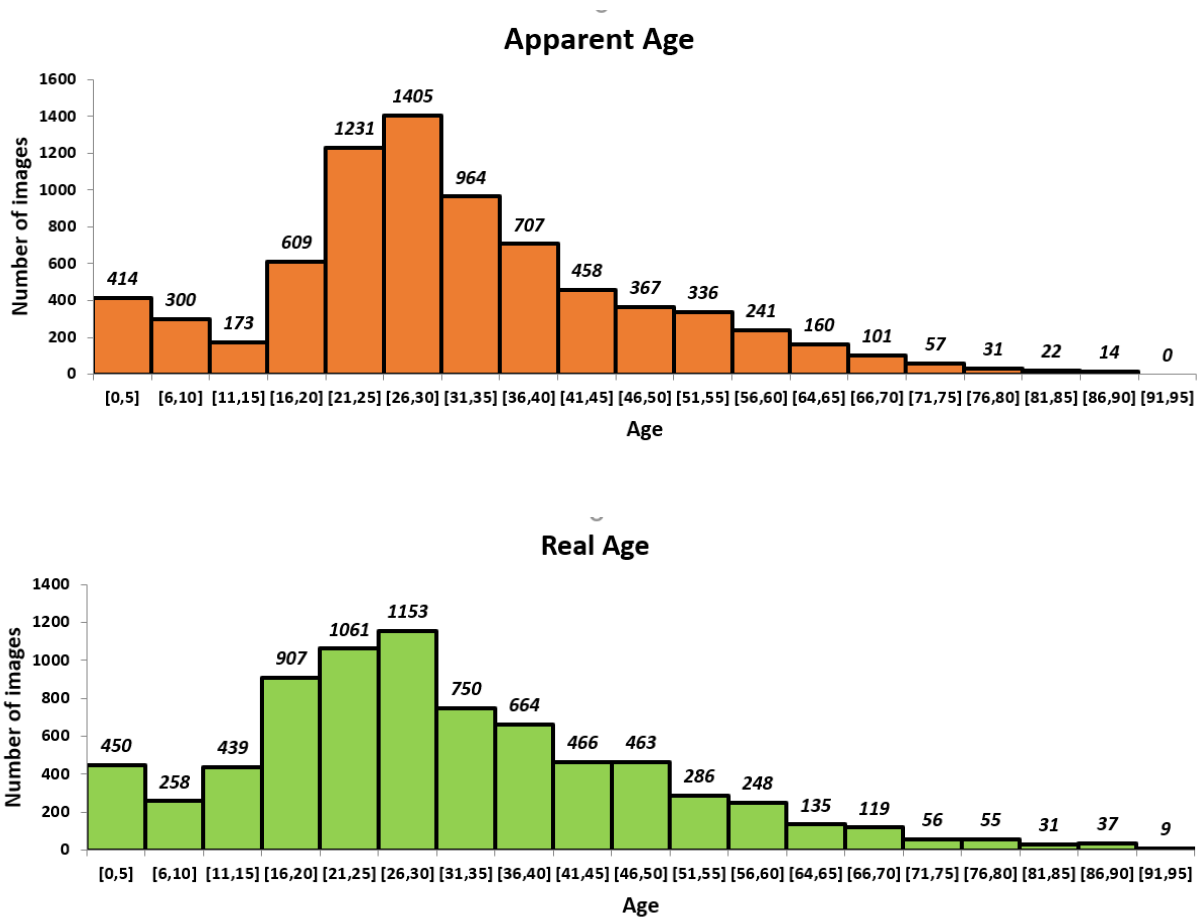


Figure 4: Age distribution in the Dataset, y-axis represents the age, and x-axis represent the number of images in the period of age

The real age and apparent age ratings are provided in the files $gt - train.csv$, $gt - test.csv$ and $gt - valid.csv$, with a separate row for each rating[6].

4.2 Data Preparation

Based on the data decription section above, I did some data preparations as shown below.

4.2.1 Dataset Split

In order to find a better split of the data, I split the data into trainset and testset for 3 times, with the ratio of 2.04:1, 2.8065:1 and 4.0753:1. The distributions of these three ways of split is shown below. I also uploaded all the data in the link: "$https : //drive.google.com/open?id = 1JioCcPiDriYt - hsWVW32HIkpmsaPUU2l$"

As for the ratio is 2.04:1, there are 5113 images split into trainset, and 2500 images split into testset. The distribution of the ages in the dataset is shown below.

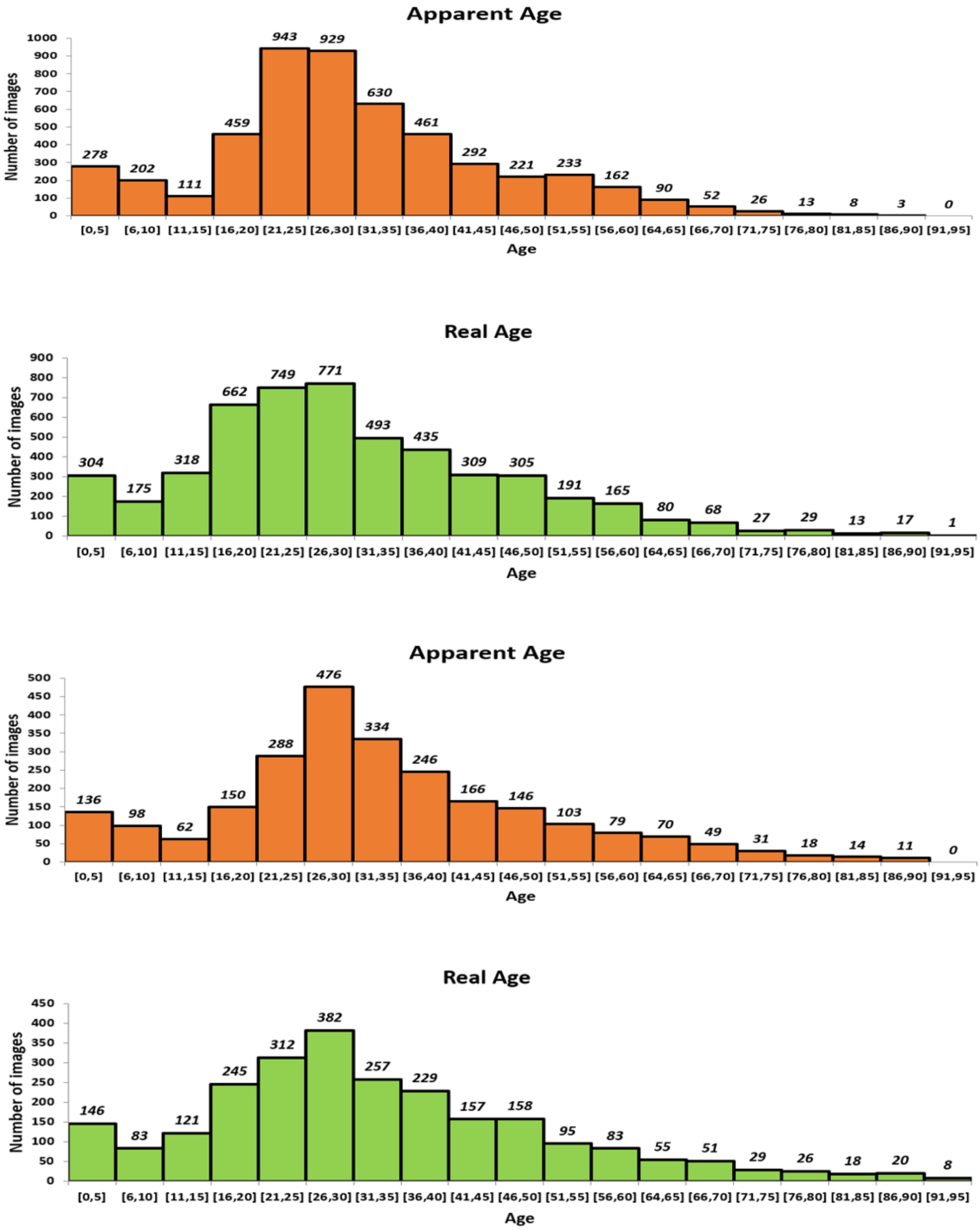


Figure 5: Age distribution. The first Two figures are for trainset with 5113 images and the last two figures are for testset with 2500 images. The y−axis represents the age, and x−axis represent the number of images in the period of age

As for the ratio is 2.8065:1, there are 5613 images split into trainset, and 2000 images split into testset. The distribution of the ages in the dataset is shown below.

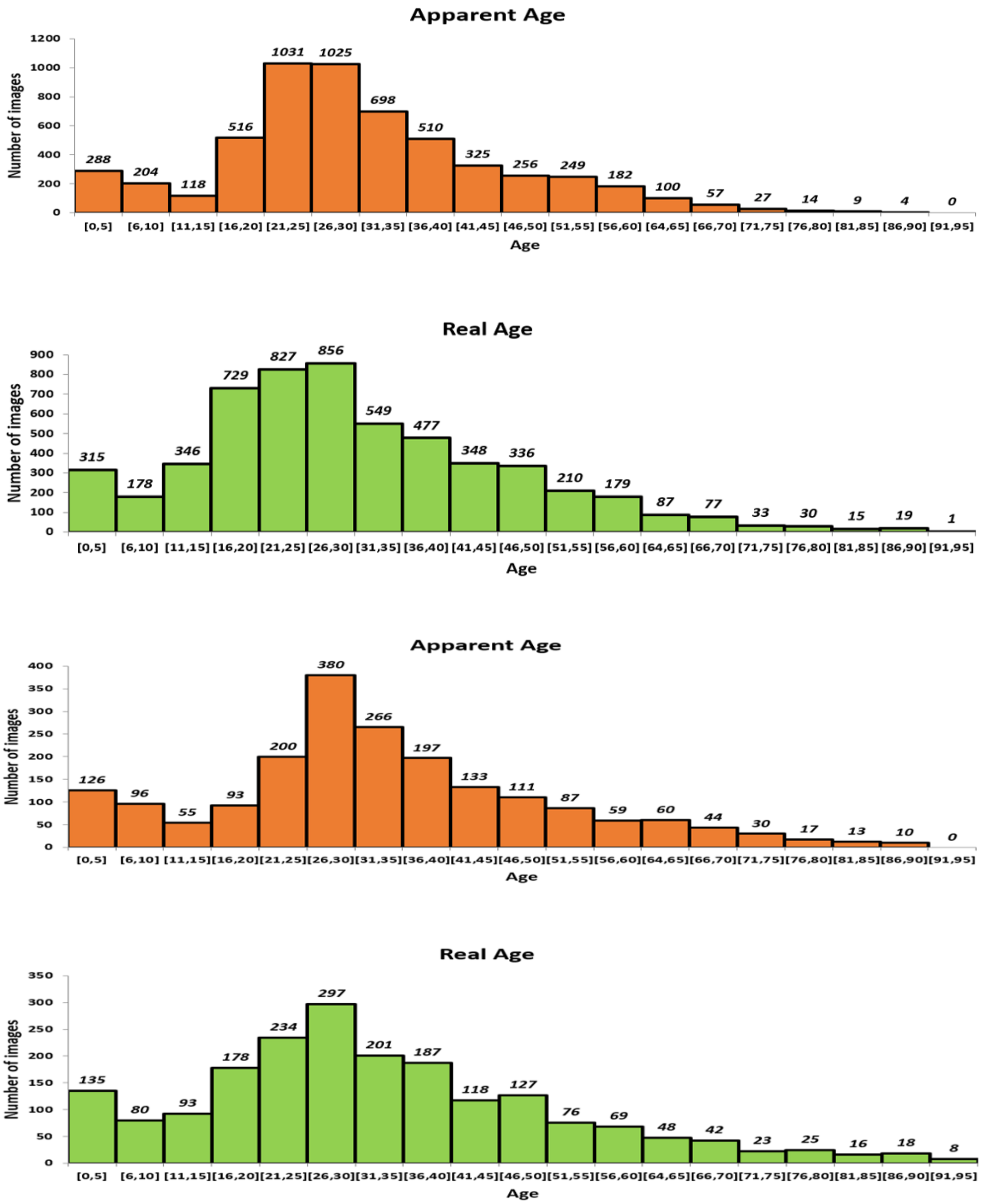


Figure 6: Age distribution. The first Two figures are for trainset with 5613 images and the last two figures are for testset with 2000 images. The y−axis represents the age, and x−axis represent the number of images in the period of age

As for the ratio is 4.0753:1., there are 6113 images split into trainset, and 1500 images split into testset. The distribution of the ages in the dataset is shown below.

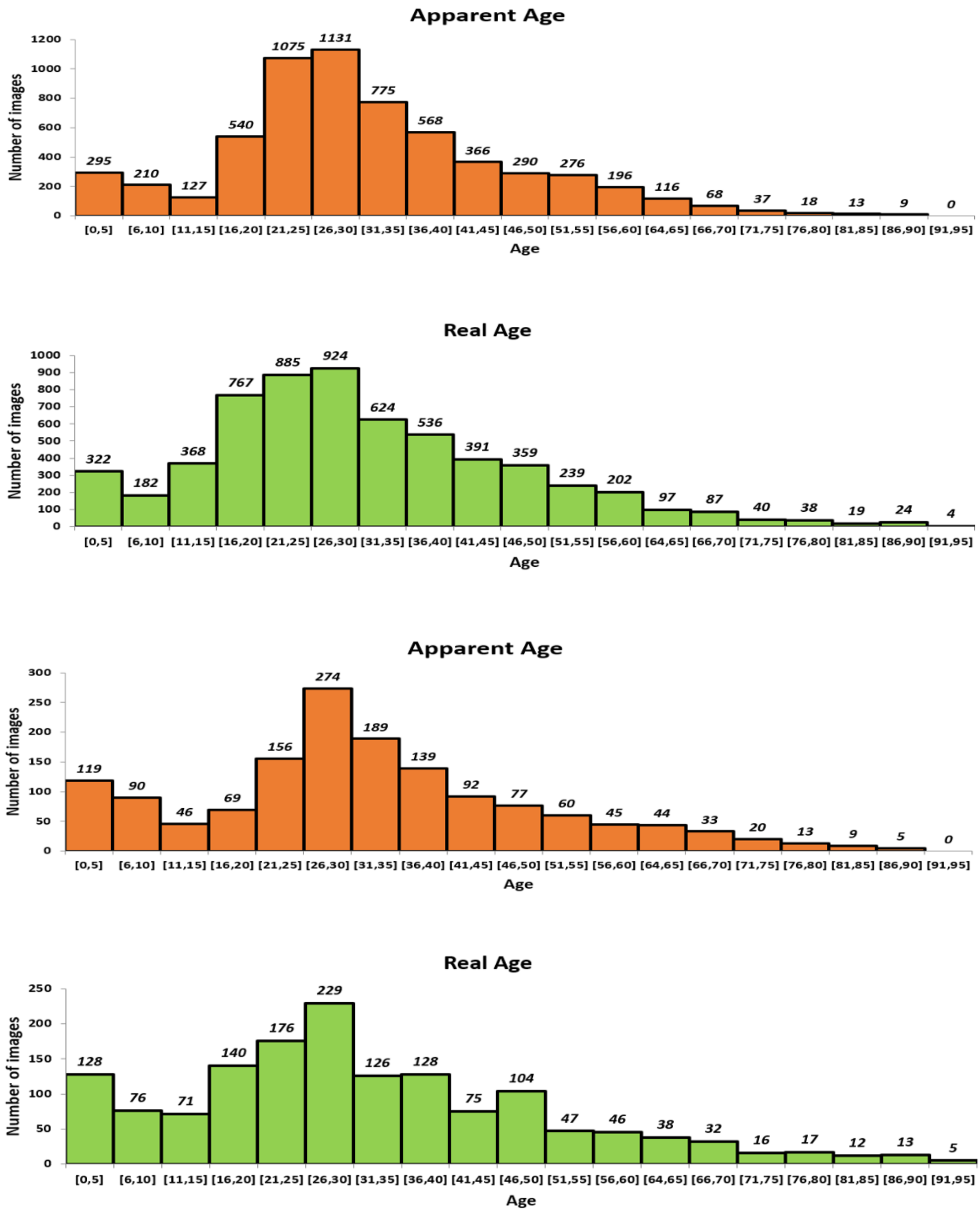


Figure 7: Age distribution. The first Two figures are for trainset with 6113 images and the last two figures are for testset with 1500 images. The y−axis represents the age, and x−axis represent the number of images in the period of age

All the split ways of the dataset offer enough The dataset I use in the term−project is APPA-REAL dataset are large enough since the dataset has been used for the competition and research on this topic, which is shown in the paper [5][6].All the split ways of the dataset offer enough trainset for the project. Since the parameters of my model must be less than 100, which makes the number of the images in trainset at least 10 times larger than the number of parameters in the model. This satisfies the "The rule of 10"[7].

4.2.2 Data Augmentation

In order to get higher accuracy of our models, I use some basic transformation methods, including transition, rotation, and scaling. The rotation degree is from -20 degrees to 20 degrees, the scaling multiples are from 0.95 to 1.05 for both x coordinate and y coordinate. The transition percent of x coordinate and y coordinate are from -0.05 from 0.05. I only introduce the method in trainset.

5 Methods

Basically, regarding to the paper[1] and the paper "Age and Gender Classification using Convolutional Neural Networks" [5], I can treat the traditional regression problem, estimation of the age, as a regression problem. While if I directly train a CNN model for regression problem, since the last layer is only one output layer with the large gradients, the predicted results will be unstable and has big errors.

I first trained a CNN model for classification problem with 101 classes, which represent 101 ages from 0 to 101. I have tried 96 classes, while it is less accurate than the model for 101 classes. Then I computed the expect value of the output of the softmax normalized layer to get the predicted age.

In my term project, I built my own models based on the CNN architecture using Pytorch library and python language, together with transfer learning, data augmentation and downsampling method.

After finishing setting up the environment on the server and do more data augmentation on the training set, which laid the roots for my deeper research in the next following steps, I built five models, including Simple CNN model, General Residual Model, Modified Residual Model, Gernal Residual + Transfer learning mdoel and Modified Residual + Transfer learning model. Regarding to Deep EXpectation method[1], I directly treat VGG−16 used in the orginal paper and se_resnet50 which is used in paper[2] as the baseline models. More details will be explained as shown below.

5.1 Simple-CNN Models

The Simple−CNN model is the CNN model with 12 layers, except input layer. The architecture and parameters of the model are shown below.

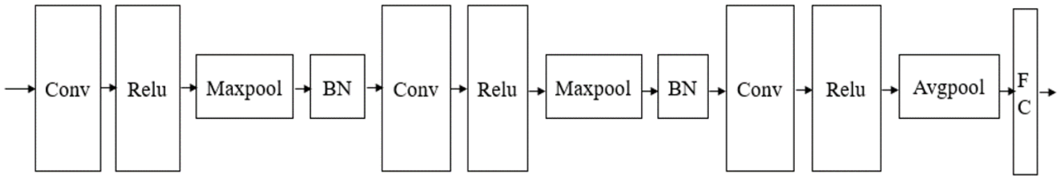


Figure 8: Architecture of the Simple-CNN model

Layer	*input*	*output*	*kernel size*	*stride*	*padding*	*dilation*	*momentum*
Cov1	**3**	**96**	**(7,7)**	**(4,4)**	\	\	\
Relu1	\	\	\	\	\	\	\
Maxpool1	\	\	**(3,3)**	**(2,2)**	**0**	**1**	\
Bn1	**96**	\	\	\	\	\	**0.1**
Conv2	**96**	**256**	**(5,5)**	**(1,1)**	**(2,2)**	\	\
Relu2	\	\	\	\	\	\	\
Maxpool2	\	\	**(3,3)**	**(2,2)**	**0**	**1**	\
Bn2	**256**	\	\	\	\	\	**0.1**
Conv3	**256**	**384**	**(3,3)**	**(1,1)**	**(1,1)**	\	\
Relu3	\	\	\	\	\	\	\
Avg_Pool	\	**(1,1)**	\	\	\	\	\
Fc1	**384**	**101**	\	\	\	\	\

Figure 9: Parameters of the SimpleCNN Model

5.2 Residual-Type Models

Regrading to the paper "Age and Gender Classification using Deep Neural Networks"[7], since it was helpful to use downsampling technique on the model for age estimation problem, I chose the residual network as the idea to this problem.

5.2.1 Residual Blocks

I built two residual blocks, including General residual block and Modified residual block which are called as layers in the following report. Both of these two blocks are based on the basic block of the residual model. All the architectures are shown below.

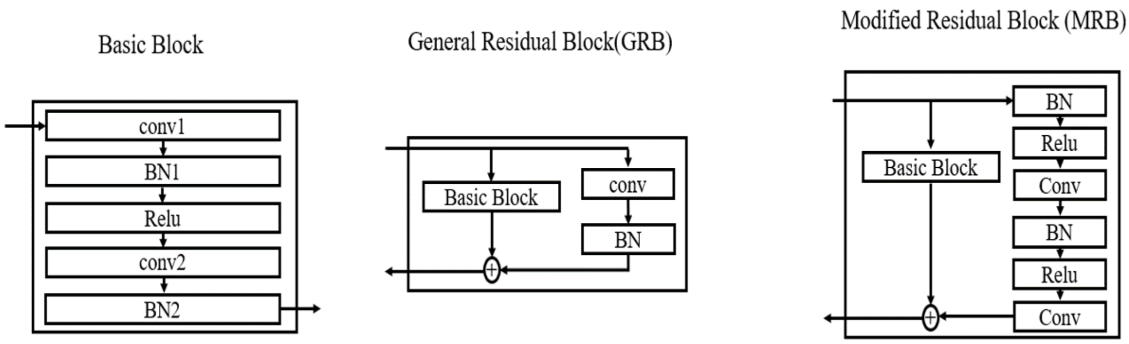


Figure 10: Resnet Block for the model

The basic block for the residual model has five layers, including two convolution layers, two batch normalization layers and one Relu layer. In the table, "sublayers in the basic block" means the layers in the BB block. For the General Residual block(GRB), there is a downsampling branch, while I modify the downsampling branch as shown in the Modified Residual Block (MRB). The parameters of these three blocks are shown above.

Layers	input	output	Kernel size	stride	padding	dilation	momentum
BB	Ch-in	Ch-out	\	s	\	\	\
Sublayers in the Basic Block							
conv1	Ch-in	Ch-out	(3,3)	s	(1,1)	\	\
bn1	Ch-out	\	\	\	\	\	0.1
relu	\	\	\	\	\	\	\
conv2	Ch-out	Ch-out	(3,3)	(1,1)	(1,1)	\	\
bn2	Ch-out	\		\	\	\	0.1

Figure 11: Parameters of the basic block used in the Residual model

In the models above, Ch−in and Ch−out represent the number of the input channels and the number of the output channels. There are 8 layers for General Residual block and 11 layers for Modified Residual block.

Layers	input	output	Kernel size	stride	padding	dilation	momentum
GRB	Ch-in	Ch-out	\	stride	\	\	\
Sublayers in the General Residual Block							
BB	Ch-in	Ch-out	\	stride	\	\	\
conv	Ch-in	Ch-out	(1,1)	stride	\	\	\
bn	Ch-out	\		\	\	\	0.1

Figure 12: Parameters of the General Residual Block

Layers	input	output	Kernel size	stride	padding	dilation	momentum
MRB	Ch-in	Ch-out	\	stride	\	\	\
Sublayers in the Modified Residual Block							
BB	Ch-in	Ch-out	\	stride	\	\	\
BN1	Ch-in	Ch-out	(1,1)	stride	\	\	\
Relu	Ch-out	\		\	\	\	0.1
conv1	Ch-in	Ch-out	(1,1)	stride	\	\	\
bn2	Ch-out	\		\	\	\	0.1
Relu	Ch-out	\		\	\	\	0.1
conv2	Ch-in	Ch-out	(1,1)	stride	\	\	\

Figure 13: Parameters of the Modified Residual Block

5.2.2 Residual Models

Based on the General Residual block and Modified Residual block above, I can build two different models. The architectures and parameters are shown below.

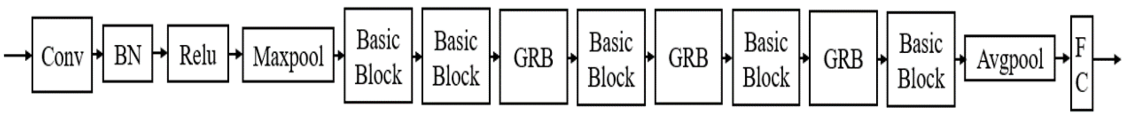


Figure 14: Architecture of General Residual model

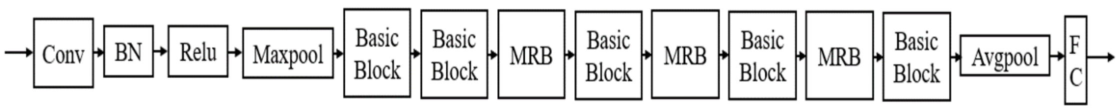


Figure 15: Architecture of Modified Residual model

The General Residual model has 62 layers except the input layer, and the Modified

Residual model has 74 layers except the input layer. The parameters of these two models are shown below.

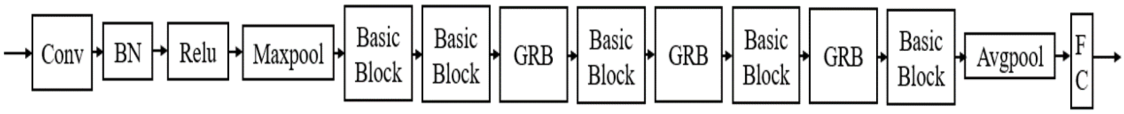


Figure 16: Architecture of General Residual model

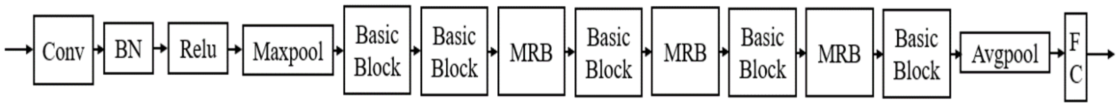


Figure 17: Architecture of Modified Residual model

Layers	input	output	kernel_size	stride	padding	dilation	momentum
Cov0	3	64	(7,7)	(2,2)	(3,3)	\	\
Bn0	64	\	\	\	\	\	0.1
Relu0	\	\	\	\	\	\	\
Maxpool0	\	\	(3,3)	(2,2)	(1,1)	1	\
BB1	64	64	\	(1,1)	\	\	\
BB2	64	64	\	(1,1)	\	\	\
GRB1/MRB1	64	128	\	(2,2)	\	\	\
BB3	128	128	\	(1,1)	\	\	\
GRB2/MRB2	128	256	\	(2,2)	\	\	\
BB4	256	256	\	(1,1)	\	\	\
GRB3/MRB3	256	512	\	(2,2)	\	\	\
BB5	512	512	\	(1,1)	\	\	\
Avg_Pool	\	(1,1)	\	\	\	\	\
Fc1	512	101	\	\	\	\	\

Figure 18: Parameters of the General Residual model and the Modified Residual model. Choose GRB in all the GRB/MRB layer for the General Residual model, while choose MRB for Modified Residual model

In figure 18, the difference between the General Residual model and the Modified Residual model is only for the GRB/MRB layer(block).

Based on the Residual models in last section, I also introduce the transfer learning technique with these two models. I substitute the initial parameters for the layers in the residual models with the parameters in RESNET18 model pretrained on imagenet. Then I can get two new models named General Residual + Transfer model and Modified Residual + Transfer model.

5.3 Quantitative Validation Method

I use the error to evaluate the model performance on specific cases, and mean absolute error (MAE) to evaluate the performance of the models.

I didn't use the accuracy as the evaluation metrics for this project. Age estimation problem is aimed to estimate the age of the person based on the image, and even human in the real world cannot easily estimate the exact number of the person based on the face information. The accuracy is calculated by comparing the predicted age with the ground truth for each of the images. But as for the apparent age labels, they are created by voting from a group people, which means you if the ground truth is 50.1, you have to get the exact same number of it. Comparing to the exact age, MAE is more meaningful.

The error is the difference between the predicted age and the ground truth age, as shown in the formula(1).

$$Error_i = |y_i - x_i| \tag{1}$$

Based on the equations above, it is easy to know that the smaller Error is, the better performance the model is on the case.

The MAE is the average of the absolute error between the predicted age and the ground truth age, as shown in the formula(2) and (3) below.

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} \tag{2}$$

$$MAE = \frac{\sum_{i=1}^{n} |actual_1 - predicted_1| + ... + |actual_n - predicted_n|}{n} \tag{3}$$

Based on the equations above, it is easy to observe that the smaller MAE is, the better performance the model is.

6 Results and Discussions

In this section, I will show the results of my 5 models and 2 baseline models. And the comparison of these 7 models. The results will based on 3 times dataset splits. All the training learning rate is 0.001, and the batch size is 64 for both training and testing. The number of epochs is 80.

6.1 Baseline models

Based on EXpectation method, I treat VGG-16 and se_resnet50 model as the baseline models. Since the vgg−16 in my code cannot run as good as the paper stated, I directly use the results of vgg−16 as the baseline results. While I tested se_resnet50 model for 3 datasets, and the results and the discussions are shown below.

Network	MAE on Apparent age				MAE on Real age			
	Trainset		Testset		Trainset		Testset	
Se_resnet50 with split method 1	4.121		5.485		0.568		8.178	
Se_resnet50 with split method 2	4.224		5.723		0.353		8.545	
Se_resnet50 with split method 3	4.381		5.328		0.425		7.876	
Se_resnet50	Trainset		Testset		Trainset		Testset	
	mean	std	mean	std	mean	std	mean	std
	4.242	0.185	5.512	0.281	0.449	0.155	8.2	0.474

Figure 19: Results for se_resnet50 model for apparent age and real age on 3 method of data split. data split method 1 is for 2.04:1; data split method 2 is for 2.8065:1 and data split method 3 is for 4.0753:1.

Based on the results shown above, the std of the model se_resnet50 is pretty small, and I treat the mean value of the MAE with three data split methods as the performance of the model. Then I can get the results below.

Network	MAE on Apparent age		MAE on Real age	
	Trainset	Testset	Trainset	Testset
VGG-16		6.47		7.61
Se_resnet50	4.242	5.512	0.449	8.2

Figure 20: Results for two baseline models; The result of VGG−16 is directly from the paper, and I choose the mean of the results of the se_resnet50 model tested with different dataset split method as the results of the se_resnet50

6.2 Simple-CNN model

I tested the Simple-CNN model with 3 data split methods, and the results and data analysis are shown below.

Network	MAE on Apparent age				MAE on Real age			
	Trainset		Testset		Trainset		Testset	
Simple-CNN with split method 1	8.594		9.736		9.259		11.466	
Simple-CNN with split method 2	8.421		10.173		9.396		12.034	
Simple-CNN with split method 3	8.483		9.893		9.086		11.691	
Simple-CNN	Trainset		Testset		Trainset		Testset	
	mean	std	mean	std	mean	std	mean	std
	8.499	0.124	9.934	0.313	9.247	0.220	11.73	0.405

Figure 21: Results for Simple-CNN model for apparent age and real age on 3 method of data split. data split method 1 is for 2.04:1; data split method 2 is for 2.8065:1 and data split method 3 is for 4.0753:1.

The model didn't perform well on both of the apparent age estimation and real age estimation.

6.3 Residual-Type Models

In this section, I will show the results for all four residual−type models, including General Residual Model, Modified Residual Model, General Ressidual + Transfer Model, and Modified Residual + Transfer Model.

6.3.1 General Residual Model

I tested the General Residual model with 3 data split methods, and the results and data analysis are shown below.

Network	MAE on Apparent age				MAE on Real age			
	Trainset		Testset		Trainset		Testset	
General Residual Model with split method 1	7.588		8.854		9.057		11.188	
General Residual Model with split method 2	7.993		9.507		8.546		11.318	
General Residual Model with split method 3	8.029		9.315		8.296		10.815	
General Residual Model	Trainset		Testset		Trainset		Testset	
	mean	std	mean	std	mean	std	mean	std
	7.870	0.346	9.225	0.475	8.633	0.549	11.107	0.369

Figure 22: Results for General Residual model for apparent age and real age on 3 method of data split. data split method 1 is for 2.04:1; data split method 2 is for 2.8065:1 and data split method 3 is for 4.0753:1.

6.3.2 Modified Residual Model

I tested the Modified Residual model with 3 data split methods, and the results and data analysis are shown below.

Network	MAE on Apparent age				MAE on Real age			
	Trainset		Testset		Trainset		Testset	
Modified Residual Model with split method 1	8.020		9.220		9.227		11.344	
Modified Residual Model with split method 2	8.012		9.626		9.803		12.138	
Modified Residual Model with split method 3	8.114		9.353		8.717		11.141	
Modified Residual Model	Trainset		Testset		Trainset		Testset	
	mean	std	mean	std	mean	std	mean	std
	8.049	0.080	9.400	0.293	9.249	0.768	11.541	0.745

Figure 23: Results for Modified Residual model for apparent age and real age on 3 method of data split. data split method 1 is for 2.04:1; data split method 2 is for 2.8065:1 and data split method 3 is for 4.0753:1.

6.3.3 General Residual + Transfer Model

Since the result is not that good, we use the transfer learning. I tested the General Residual model with 3 data split methods together with transfer learning technique, and the results and data analysis are shown below.

Network	MAE on Apparent age				MAE on Real age			
	Trainset		Testset		Trainset		Testset	
General Residual + Transfer Model with split method 1	5.260		6.380		4.778		9.603	
General Residual + Transfer Model with split method 2	6.210		7.895		5.128		10.694	
General Residual + Transfer Model with split method 3	5.744		6.936		4.988		9.375	
General Residual + Transfer Model	Trainset		Testset		Trainset		Testset	
	mean	std	mean	std	mean	std	mean	std
	5.738	0.672	7.070	1.084	4.965	0.249	9.891	0.997

Figure 24: Results for General Residual + Transfer model for apparent age and real age on 3 method of data split. data split method 1 is for 2.04:1; data split method 2 is for 2.8065:1 and data split method 3 is for 4.0753:1.

It is easy to find that there are huge improvement comparing to the original General Residual model.

6.3.4 Modified Residual + Transfer Model

Since the result is not that good, we use the transfer learning. I tested the Modified Residual model with 3 data split methods together with transfer learning technique, and the results and data analysis are shown below.

Network	MAE on Apparent age				MAE on Real age			
	Trainset		Testset		Trainset		Testset	
Modified Residual + Transfer Model with split method 1	5.412		6.830		5.509		8.942	
Modified Residual + Transfer Model with split method 2	5.829		7.150		6.222		9.785	
Modified Residual + Transfer Model with split method 3	5.687		6.483		5.718		8.830	
Modified Residual + Transfer Model	Trainset		Testset		Trainset		Testset	
	mean	std	mean	std	mean	std	mean	std
	5.643	0.300	6.821	0.472	5.816	0.518	9.186	0.738

Figure 25: Results for Modified Residual + Transfer model for apparent age and real age on 3 method of data split. data split method 1 is for 2.04:1; data split method 2 is for 2.8065:1 and data split method 3 is for 4.0753:1.

It is easy to find that there are huge improvement comparing to the original Modified Residual model.

6.4 Comparison of different models

I put all the results I get from the formal sections in one table shown below to do the comparison. As mentioned in the method chapter, the smaller MAE is, the better the model is.

Network	MAE on Apparent age		MAE on Real age	
	Trainset	Testset	Trainset	Testset
VGG-16		6.47		**7.61**
Se_resnet50	4.242	**5.512**	0.449	8.2
Simple CNN model	8.499	9.934	9.247	11.73
General Residual Model	7.870	9.225	8.633	11.107
Modified Residual Model	8.049	9.400	9.249	11.541
General Residual + Transfer Model	5.738	7.070	4.965	9.891
Modified Residual + Transfer Model	5.643	***6.821***	5.816	***9.186***

Figure 26: The comparison of the models

In the table, the best model is the se_resnet50 for apparent age estimation, while the VGG−16 is the best for real age estimation. The best model of mine is Modified Residual + Transfer Model. The performance of it is also great when estimating on apparent age, since it almost closed to the performance of VGG−16. While as for the real age estimation, it doesn't perform well. From my point of view, in our daily life, when we guess someone's age only based on the face information, what we guess is the apparent age. Someone with good condition of the face may confuse the people about their age. So does the model we built. So we may introduce other features, such as positions, emotions, when we are doing real age estimation.

Anyway, with my efforts, the performance of my model is getting better, which support the correctness of my methods.

6.5 Other results and discussions for my best model

In the former section, it is easy to find that my best model is Modified Residual + Transfer model. In this section, there shows some other results for the model, including feature map and some Representative examples of apparent and real age estimations.

I get the feature map for each layers for a image, and the predicted age is shown. In this case, the error is only 0.08 for the apparent age estimation.

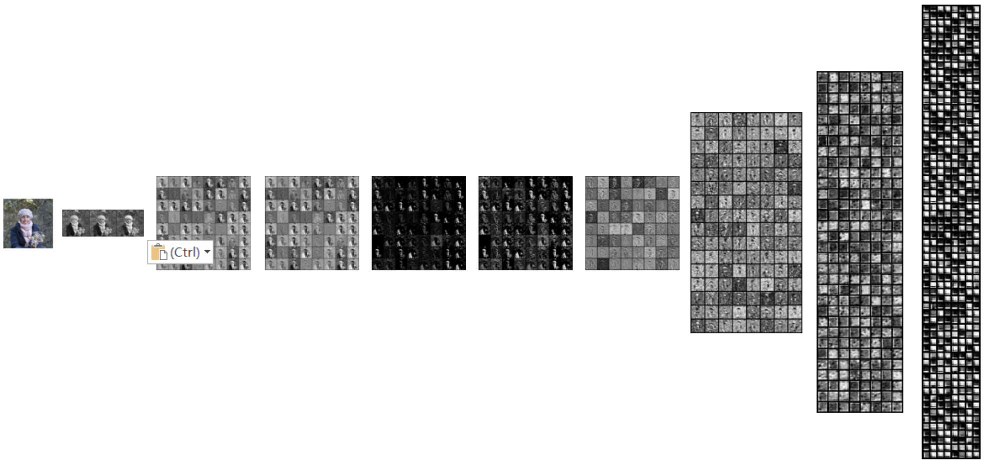


Figure 27: The feature map of each layer for one case

Figure 28: The output of the estimation results on this image

Apparent Age	Ground Truth	43.92
	Predicted Age	44

Figure 29: The comparison of the Ground Truth and Predicted Age

The feature map for each layers for a image of real age estimation, and the predicted age is shown. In this case, the error is 5 for the real age estimation.

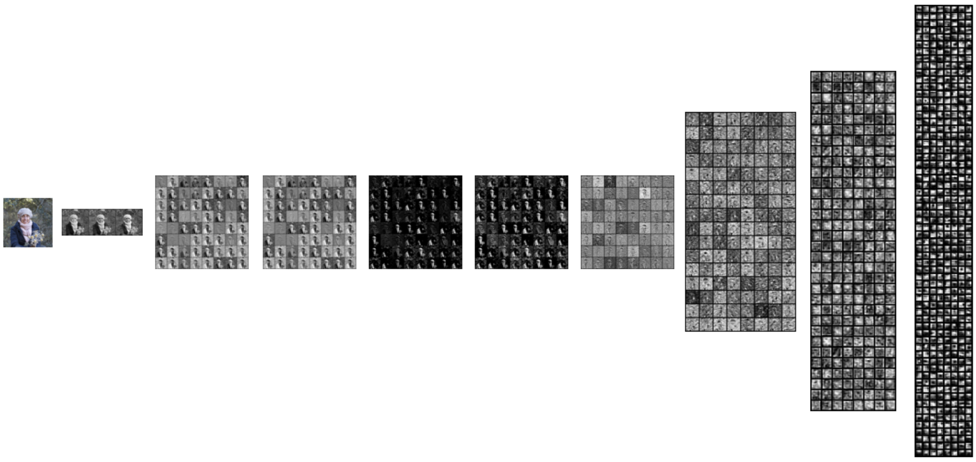


Figure 30: The feature map of each layer for one case

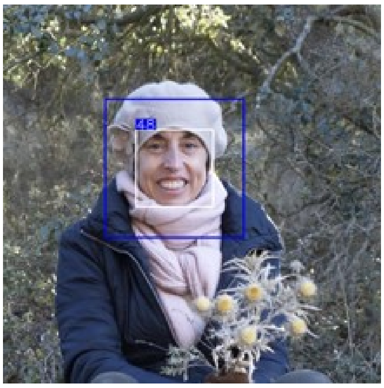


Figure 31: The output of the estimation results on this image

Real Age	Ground Truth	53
	Predicted Age	48

Figure 32: The comparison of the Ground Truth and Predicted Age

When using the Modified Resudial + Transfer Model on some cases in the test set, the results are shown below. In the figure 33, GT means Ground Truth, and PA means predicted results.

Apparent Age	GT	5	5.81	26.83	28.03	40.40	41	42.92	43.92	63.97	76.82
	PA	6	13	25	31	11	18	36	44	59	68
Real Age	GT	4	6	32	24	47	50	35	53	58	80
	PA	9	11	29	30	9	23	54	48	59	77

Figure 33: Comparison of the ground truth and predicted age of Modified Residual + Transfer model

Based on the results above, it is easy to get that our model perform greater in the middle age group. And based on the dataset distribution, I have more data in the middle age group, which make this result reasonable. And if you look into the 5th column of the picture in figure 33, you can find that the model may fail in the case of the person with good make-up.

7 Conclusion

In the project, I built five models. The performance of the models is getting better from the Simple CNN model to the Modified Residual + Transfer model. The MAE of the Modified Residual + Transfer model on apparent age estimation is 6.821 with the testset, while it is 9.186 on real age estimation with the testset. Its MAE score for apparent age estimation is close to the baseline VGG−16, while there is still a distance between it with the performance of se_resnet50 model. Its MAE score for real age estimation is close to the baseline se_resnet50, while there is still a distance between it with the performance of VGG−16 model.

Overall, I think my methods work well, based the comparison between these 5 models and 2 baseline models, and this is in my expectation. While there are some cons of my models, which can be improved in the future(The results analysis is on the results and discussions part).

7.1 What I learned from the project

I learned three things from the project. First, I have learned how to work out the problem with real data. Second, I have learned how to come up my methods by reading the related papers. Third, I have got familiar with machine learning and deep learning, together with coding using python and the Pytorch library.

7.2 future work

Based on the result analysis in the Results and Discussions chapter, the following things can be worked on in the future. First, doing age estimation not only depending on the image and age labels, but also trying to combine other features. This is because the results of my models did well in apparent age estimation, instead of the real age estimation. The reason in details in discussed in the former chapter. Second, creating more powerful network. Although my models are great in the apparent age estimation, it still can be improved. Third, combining the traditional machine learning technique together with the deep learning method.

7.3 The timetable for each step

Here is the timetable for each of my steps for this project.

Table 1: Weekly timetable for the project. This is an example of a longtable which can automatically stretch to multiple pages.

Week	Info
Week 1 (3/15-3/21):	Prepare the data and set up the environment on the server.
Week 2 (3/22–3/28):	To understand the model and method on CNN in the papers and build my own model.
Week 3 (3/29–4/4):	Build the model to train and test my own model
Week 4 (4/5–4/11):	Working on the midterm-way report and presentation and also Working on the step 5 to modify the parameters to train and test again and again to get the performance improving
Week 5 (4/12–4/18):	Compare the results and modify the parameters to train and test again and again to get the performance improving, using transfer learning and do some fine-tune on my dataset.
Week 6 (4/19–4/25):	Compare the results and modify the parameters to train and test again and again to get the performance improving, using transfer learning and do some fine-tune on my dataset.
Week 7 (4/26–5/4):	Working on the step 5 and finishing the final report and presentation
Week 8 (5/5):	Project Due!

References

[1] L. V. G. Rasmus Rothe, Radu Timofte, "Deep expectation of real and apparent age from a single image without facial landmarks," *International Journal of Computer Vision*, vol. 126, p. 144–157, 2018.

[2] S. E. X. B. I. G. E. Agustsson, R. Timofte and R. Rothe, "Apparent and real age estimation in still images with deep residual regressors on appa-real database," *Proc. of FG*, 2017.

[3] "https://github.com/yu4u/age-gender-estimation."

[4] L. V. G. Rasmus Rothe, Radu Timofte, "Dex: Deep expectation of apparent age from a single image," *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, 2015.

[5] S. E. e. a. Albert Clapes, Gholamerza Anbarjafari, "From apparent to real age: gender, age, ethnic, makeup, and expression bias analysis in real age estimation," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 2373–2382, 2018.

[6] "Appa-real (real and apparent age)," http://chalearnlap.cvc.uab.es/dataset/26/description/.

[7] "dataset size lecture in cse583."