FaceAgeGenderRecognition

Datasets

- CelebFaces Attributes Dataset (CelebA) [1]

  A large-scale face attributes dataset with more than 200K celebrity images, each with 40 attribute annotations. The images in this dataset cover large pose variations and background clutter. In this project we only use its gender annotation for our model training, and this model we called pre-trained model is prepared for fully Age and Gender model training.

- The Asian Face Age Dataset (AFAD) [2]

  A dataset proposed for evaluating the performance of age estimation, which contains more than 160K facial images and the corresponding age and gender labels. It consists of 63,680 photos for female as well as 100,752 photos for male, and the ages range from 15 to 40.

- MegaAge-Asian [3]

  A dataset consists only Asian faces (40,000 face images) and the ages range from 0 to 70.

- UTKFace [4]

  A dataset is a large-scale face dataset with long age span (range from 0 to 116 years old). The dataset consists of over 20,000 face images with annotations of age, gender, and ethnicity. The images cover large variation in pose, facial expression, illumination, occlusion, resolution, etc.

In training steps, we first use CelebA for gender training of our model called pre-trained model, then merge two datasets include AFAD and MegaAge-Asian into Asian Dataset and use UTKFace as American-European Dataset for Age and Gender training on pre-trained model.

In image processing, we delete the images which resolution is smaller than 32×32 and over blurry, then use cvlib [5] to detect the faces in images and crop it.

*** Notice: the datasets above are all non-commercial, this project is just in development. ***

Model Training Details

- Model Structure

1. The original model is named MobileFaceNet [6], and its structure is the following:

Table 1. MobileFaceNet architecture for feature embedding. We use almost the same notations as MobileNetV2 [3]. Each line describes a sequence of operators, repeated *n* times. All layers in the same sequence have the same number *c* of output channels. The first layer of each sequence has a stride *s* and all others use stride 1. All spatial convolutions in the bottlenecks use 3 × 3 kernels. The expansion factor *t* is always applied to the input size. GDConv7x7 denotes GDConv of 7 × 7 kernels.

Input	Operator	*t*	*c*	*n*	*s*
$112^2 \times 3$	conv3x3	-	64	1	2
$56^2 \times 64$	depthwise conv3x3	-	64	1	1
$56^2 \times 64$	bottleneck	2	64	5	2
$28^2 \times 64$	bottleneck	4	128	1	2
$14^2 \times 128$	bottleneck	2	128	6	1
$14^2 \times 128$	bottleneck	4	128	1	2
$7^2 \times 128$	bottleneck	2	128	2	1
$7^2 \times 128$	conv1x1	-	512	1	1
$7^2 \times 512$	linear GDConv7x7	-	512	1	1
$1^2 \times 512$	linear conv1x1	-	128	1	1

2. The modified model for FaceAgeGenderRecognition is the following:

Input	Operator	t	c	n	s
$62^2 \times 3$	conv3x3	-	48	1	2
$31^2 \times 48$	depthwise conv3x3	-	48	1	1
$31^2 \times 48$	bottleneck	2	48	5	2
$16^2 \times 48$	bottleneck	4	96	1	2
$8^2 \times 96$	bottleneck	2	96	6	1
$8^2 \times 96$	bottleneck	4	96	1	2
$4^2 \times 96$	bottleneck	2	96	2	1
$4^2 \times 96$	conv1x1	-	512	1	1
$4^2 \times 512$	**linear GDConv 4x4**	-	512	1	1
$1^2 \times 512$	linear conv1x1	-	128	1	1

A GDConv layer is a depthwise convolution layer with kernel size equalling the input size, pad = 0, and stride = 1.

A bottleneck is combined by depthwise convolution and pointwise convolution with expansion layer. Expansion layer will increase the channel k of input to tk.

The input size of face images is 62 × 62. The width of modified model is different from the original because we fine-tune the width by multiplying 0.75 for every layer. Through this way we can decrease both parameter and GFLOPs to increase speed on real-time recognition and keep the high accuracy.

- Training Details

1. We use pre-trained model training from CelebA dataset with Gender and expand it by adding a branch for Age training. The Age and Gender both are the classification, mission of this project. Gender has two classes called male (label 1) and female (label 0), then we have 48 classes in 18~65 years old for training Age recognition.

2. By Age and Gender training process, the hyperparameter is described by the following:

   Batch size: 128
   Optimizer: Adam
   Learning rate: 0.01 with no decay
   Age loss: SparseCategoricalCrossentropy in tf.keras
   Gender loss: CategoricalCrossentropy in tf.keras
   Training step: input 128 samples and calculate the average loss and gradient is one step, finally the model needs 1270000 steps for training totally.

3. In order to deal with the unbalance of age distribution, we adjust the composition of the batch by the following:

   i. First, we choose to put two years apart in a group like (18~20), (21~23), (24~26) and so on. Because the similarity of feature in a single group, we think it will beneficial for model training. Finally, we get 16 groups in 48 classes.
   ii. Second, we split the 128 samples to half. One half is for Asian and another half is for American, European and so on. In 64 samples of Asian or American, we furthermore assign different batch in every groups as the following describe.

		Initial Training	
		Asian Batch	American and European Batch
Group	(18~20)	6	5
	(21~23)	6	5
	(24~26)	6	5
	(27~29)	6	5
	(30~32)	6	5
	(33~35)	6	5
	(36~38)	6	5
	(39~41)	6	5
	(42~44)	2	3
	(45~47)	2	3
	(48~50)	2	3
	(51~53)	2	3
	(54~56)	2	3
	(57~59)	2	3
	(60~62)	2	3
	(63~65)	2	3

		After 560000 Training steps	
		Asian Batch	American and European Batch
Group	(18~20)	8	6
	(21~23)	8	6
	(24~26)	8	6
	(27~29)	8	6
	(30~32)	6	6
	(33~35)	6	6
	(36~38)	6	6
	(39~41)	6	6
	(42~44)	1	2
	(45~47)	1	2
	(48~50)	1	2
	(51~53)	1	2
	(54~56)	1	2
	(57~59)	1	2
	(60~62)	1	2
	(63~65)	1	2

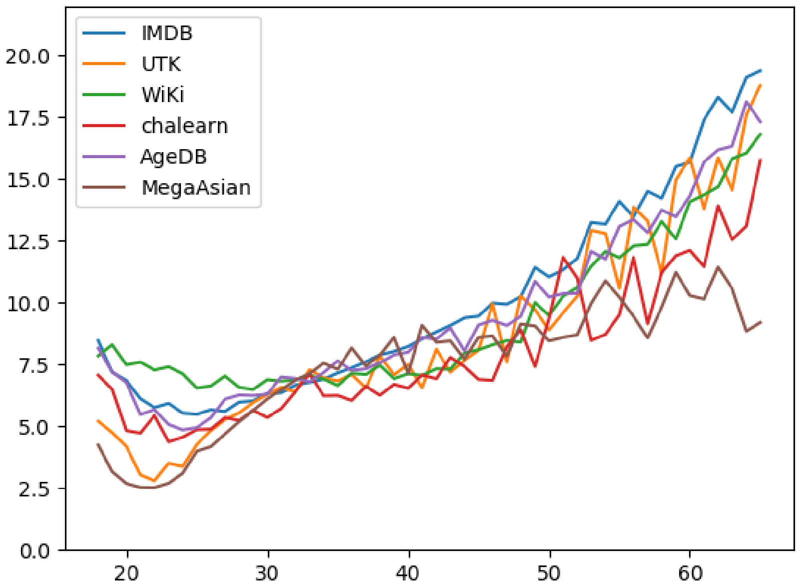
Results

- ## Gender Recognition

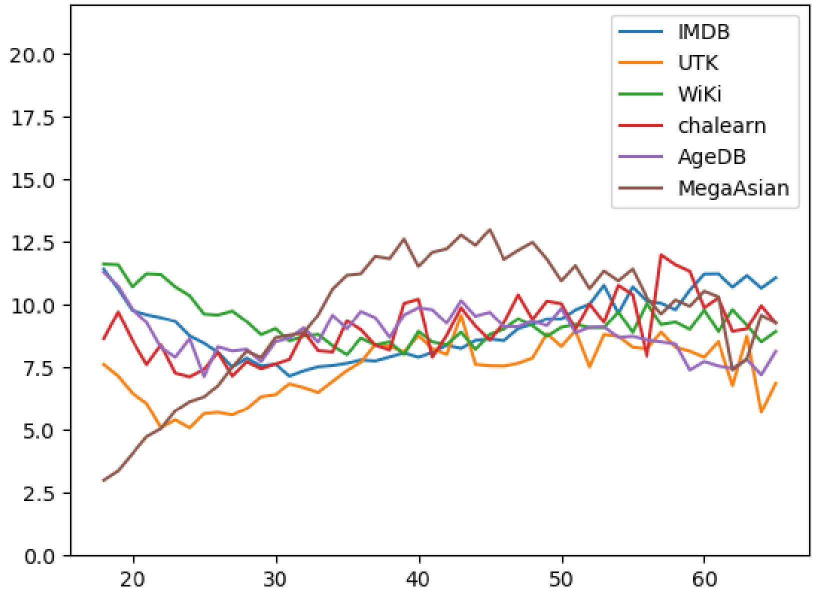
	Adience						LAG						LFW					
	Recall		Precision		F1-score		Recall		Precision		F1-score		Recall		Precision		F1-score	
	Male	Female	Male	Female	Male	Female	Male	Female	Male	Female	Male	Female	Male	Female	Male	Female	Male	Female
OpenVINO	0.687	0.891	0.845	0.766	0.758	0.824	0.705	0.965	0.957	0.747	0.812	0.842	0.942	0.922	0.977	0.822	0.959	0.869
InsightFace	0.678	0.860	0.808	0.754	0.722	0.827	0.710	0.966	0.958	0.757	0.811	0.879	0.969	0.874	0.964	0.889	0.973	0.920
Our Model	**0.696**	**0.936**	**0.904**	**0.780**	**0.732**	**0.860**	**0.701**	**0.977**	**0.971**	**0.747**	**0.805**	**0.883**	**0.983**	**0.957**	**0.988**	**0.941**	**0.980**	**0.963**

	CelebA						IMDB						WiKi					
	Recall		Precision		F1-score		Recall		Precision		F1-score		Recall		Precision		F1-score	
	Male	Female	Male	Female	Male	Female	Male	Female	Male	Female	Male	Female	Male	Female	Male	Female	Male	Female
OpenVINO	0.947	0.961	0.945	0.962	0.946	0.961	0.917	0.805	0.852	0.888	0.883	0.844	0.948	0.805	0.937	0.835	0.942	0.820
InsightFace	0.938	0.972	0.960	0.956	0.957	0.979	0.890	0.860	0.886	0.865	0.882	0.887	0.945	0.821	0.941	0.830	0.942	0.880
Our Model	**0.960**	**0.985**	**0.978**	**0.972**	**0.968**	**0.985**	**0.917**	**0.870**	**0.896**	**0.895**	**0.896**	**0.892**	**0.980**	**0.921**	**0.974**	**0.937**	**0.960**	**0.934**

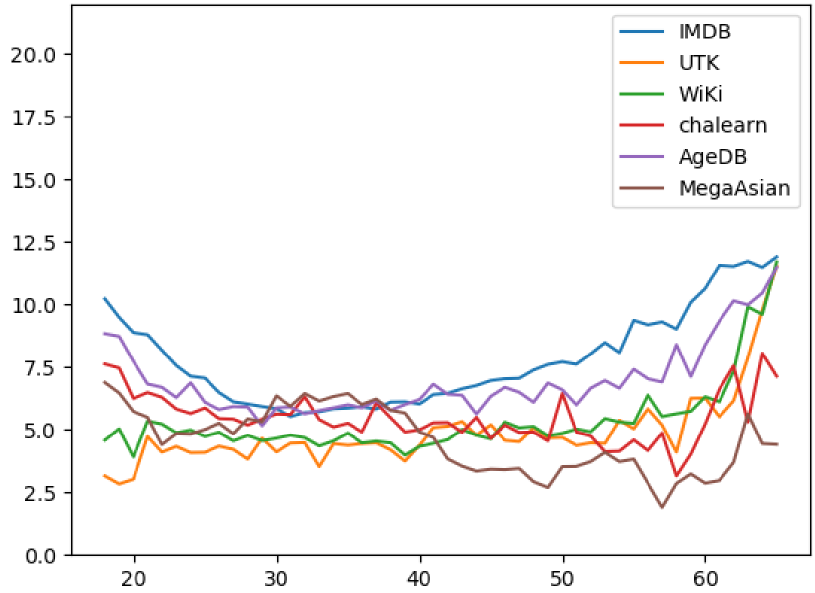
- ## Age Estimation



(a) OpenVINO



(b) InsightFace



(c) Our Model

Fig 1. Results of (a) OpenVINO, (b) InsightFace and (c) Our Model age estimation. X-axis means range of age from 18 to 65, and y-axis means the mean absolute error (MAE) of model estimation in every age.

Reference

[1] Liu, Ziwei and Luo, Ping and Wang, Xiaogang and Tang, Xiaoou, “Deep Learning Face Attributes in the Wild,” 2015.

[2] Zhenxing Niu, Mo Zhou, Le Wang, Xinbo Gao, Gang Hua, “Ordinal Regression With Multiple Output CNN for Age Estimation,” *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 4920-4928, 2016.

[3] Yunxuan Zhang, Li Liu, Cheng Li, and Chen Change Loy, “Quantifying Facial Age by Posterior of Age Comparisons,” *British Machine Vision Conference (BMVC),* 2017.

[4] Zhang, Zhifei, Song, Yang, and Qi, Hairong, “Age Progression/Regression by Conditional Adversarial Autoencoder,” *IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 2017.

[5] Ponnusamy, Arun, “cvlib - high level Computer Vision library for Python,” 2018. [Online]. Available: https://github.com/arunponnusamy/cvlib.

[6] S. Chen, Y. Liu, X. Gao, and Z. Han, “Mobilefacenets: Efficient cnns for accurate real-time face verification on mobile devices,” *arXiv preprint arXiv:1804.07573*, 2018.