Deep Learning for Face Analysis and Fine-grained Visual Recognition

Speaker： Lu Xu
Supervisor: Jinhai Xiang
Research interests: Deep Learning/Computer Vision

CONTENTS

01 Introduction

Deep Learning

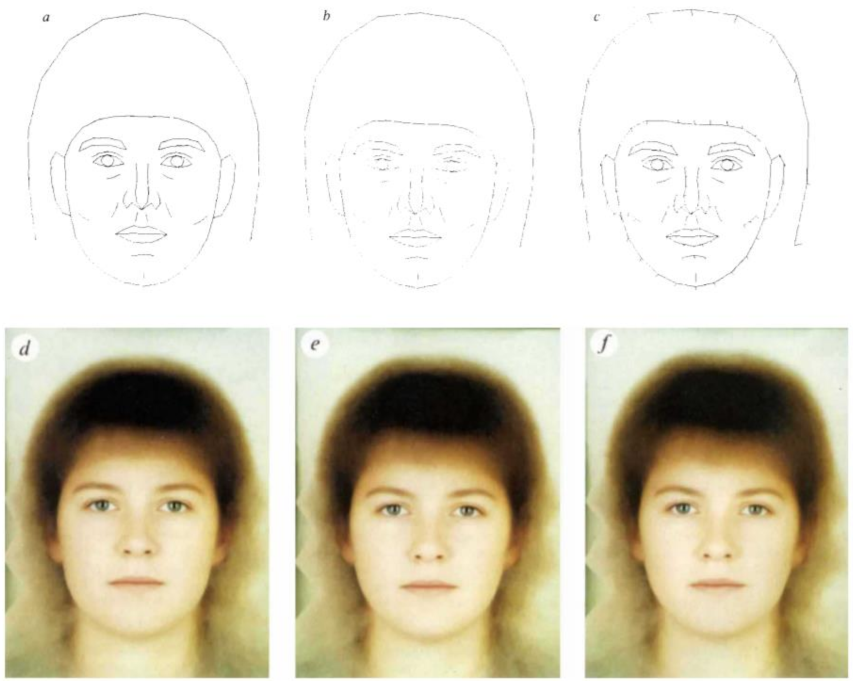
Deep learning (deep neural network) is a subset of representation learning. It can extract more abstract and more discriminative features than hand-crafted descriptors. DL has been widely used in many fields (especially AI-complete tasks like CV, NLP, and Speech).

Face Analysis

To recognize facial attributes (such as gender, race, beauty, age, expression, etc.) from a portrait image. It has been widely used among SNS and short video platforms (like TikTok, Meitu and Instagram).

01 Introduction

Facial Beauty Prediction



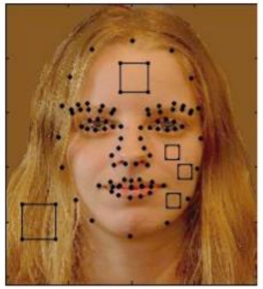


Figure 1: Facial coordinates with hair and skin sample regions as represented by the facial feature extractor. Coordinates are used for calculating geometric features and asymmetry. Sample regions are used for calculating color values and smoothness. The sample image, used for illustration only, is of T.G. and is presented with her full consent.

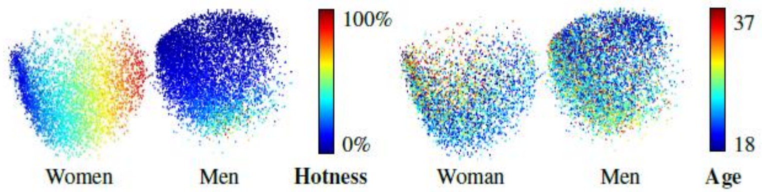


Figure 9. Visualization of latent space Q for women and men.

- **Conclusion**

Facial beauty perception is subjective in personal view, but shows stability in group.

It can be automatically predicted by data-driven models.

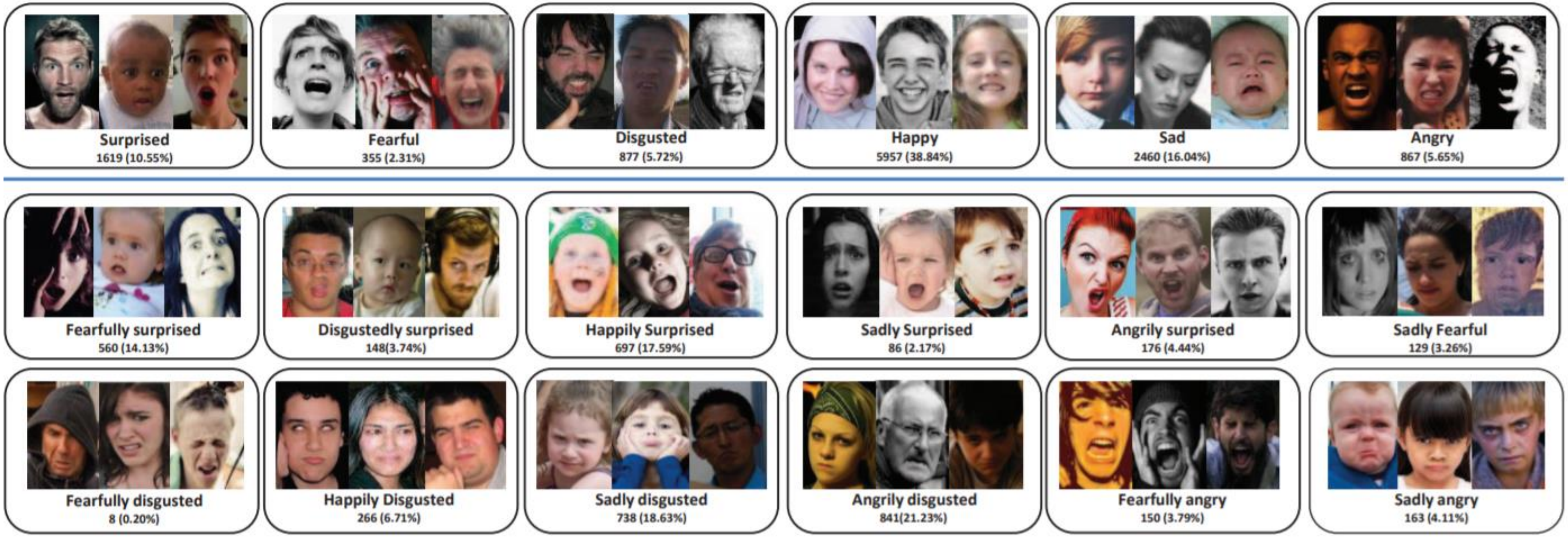
- **Tendency**

Moving from hand-crafted features to deep learning models.

1. Perrett, David I., Karen A. May, and Sin Yoshikawa. "Facial shape and judgements of female attractiveness." Nature 368.6468 (1994): 239.
2. Kagian, A., Dror, G., Leyvand, T., Cohen-Or, D., Ruppin, E.: A humanlike predictor of facial attractiveness. NIPS, pp. 649–656 (2007)
3. Rothe, R., Timofte, R., Van Gool, L.: Some like it hot-visual guidance for preference prediction. In: Proceedings CVPR 2016, pp. 1–9 (2016)

01 Introduction

Facial Expression Recognition



Developing computational models to automatically recognize a person’ s facial expression (such as happy, sad, angry, etc.)

Age Estimation

The learning algorithm should give a precise age estimation from a portrait image.

02 Deep Learning for Face Analysis

Facial Attributes Analysis

- Transferring Rich Deep Features for Facial Beauty Prediction

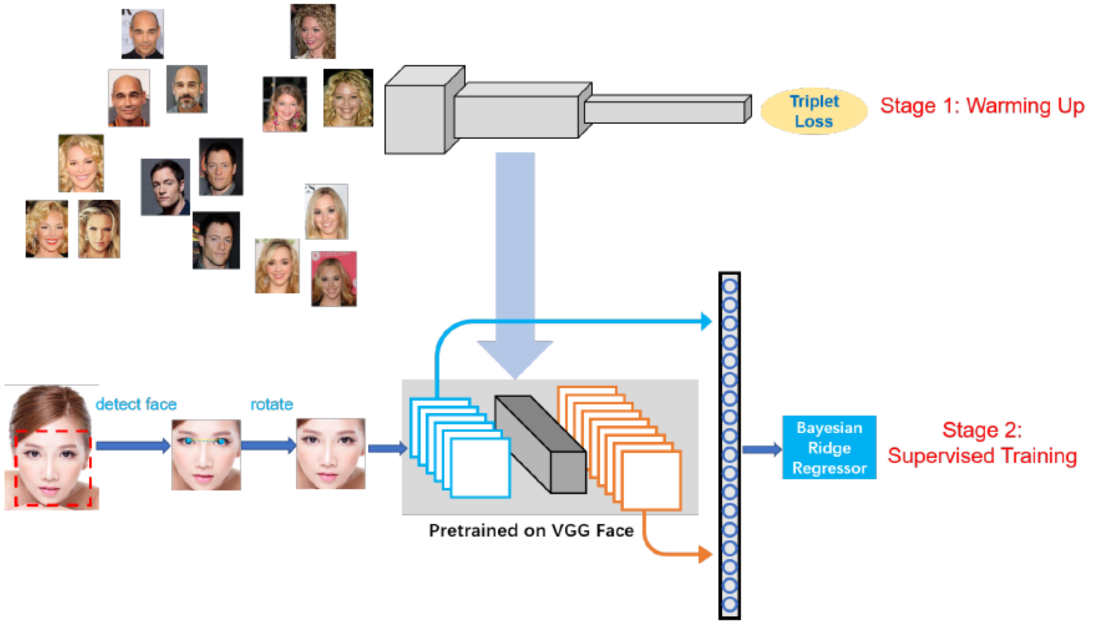


Fig. 1. Pipeline of our proposed method. We firstly train a face verification task on VGG Face dataset to obtain facial beauty representation. Then the face is detected, rotated and then fed into the pre-trained model, we concatenate both low level and high-level features for more informative facial representation, and flatten them into feature vectors for the input of Bayesian ridge regression.

$$E(W') = \sum_{(a,p,n)\in T} max\{0, \alpha - ||x_a - x_n||_2^2 + ||x_a - x_p||_2^2\}, \quad x_i = W' \frac{\phi(l_i)}{||\phi(l_i)||_2}$$

Table 3. Performance comparison with other methods. Our method achieves state-of-the-art performance on the SCUT-FBP dataset. The best performance is highlighted in bold.

Method	PC
Combined Features+Gaussian Reg [3]	0.6482
CNN-based [3]	0.8187
Liu et al. [23]	0.6938
KFME [25]	0.7988
RegionScatNet [26]	0.83
PI-CNN [11]	0.87
Ours	**0.8742**

Facial Attributes Analysis

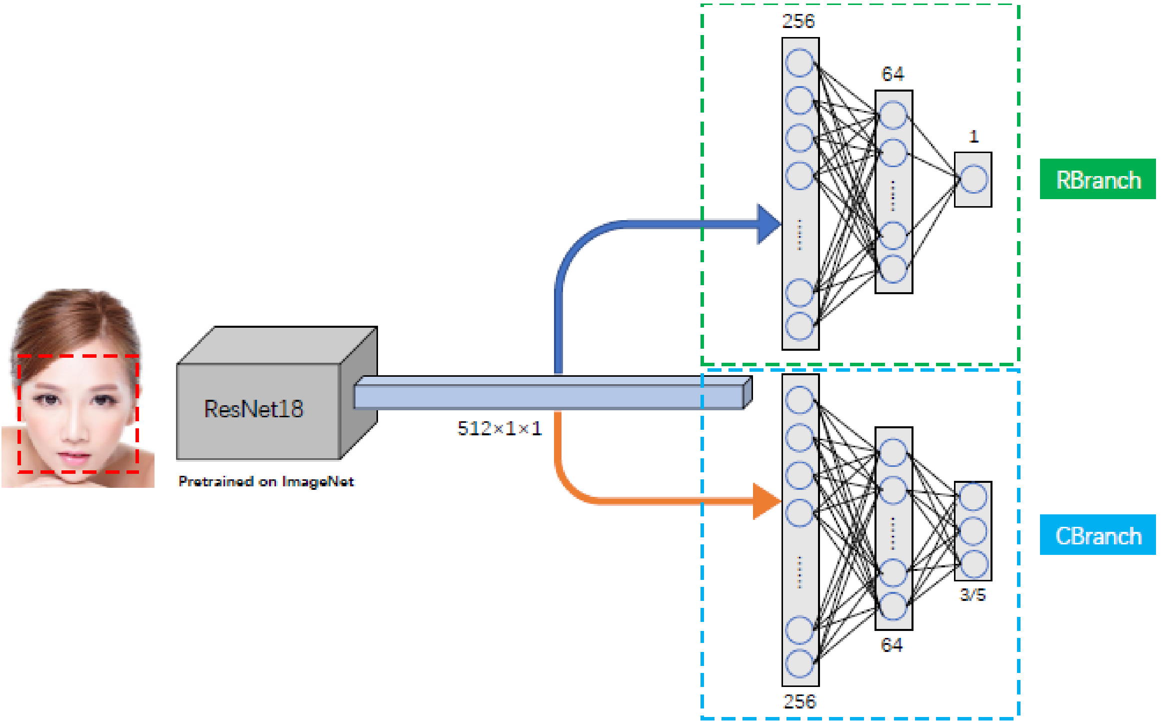
- Transferring Rich Deep Features for Facial Beauty Prediction

- Our proposed methods indicate that the model pretrained on a totally different task with different data distribution, guided by different loss function also contains informative representation for beauty.
- Our methods achieves state-of-the-art performance on relevant benchmark datasets.

So can the features be shared among different tasks?

Facial Attributes Analysis

- CRNet: Classification and Regression Neural Network for Facial Beauty Prediction (Pacific Rim Conference on Multimedia 2018)



$$\mathcal{L}_c = -\frac{1}{M}\sum_{i=1}^{M} y_i \cdot log\hat{y}_i$$

$$\mathcal{L}_r = \frac{1}{M}\sum_{i=1}^{M} (y_i - \hat{y}_i)^2$$

$$\mathcal{L} = \theta_c \cdot \mathcal{L}_c + \theta_r \cdot \mathcal{L}_r$$

$$c = \begin{cases} 0\ ; & if\ c_i < -1 \\ 1\ ; & if\ -1 \le c_i < 1 \\ 2\ ; & otherwise \end{cases}$$

Table 3. Performance comparision with baseline models on ECCV HotOrNot dataset

Method	PC
Multiscale Model [5]	0.458
S. Wang et al. [12]	0.437
CRNet	**0.482**

Performance comparison with recent baseline models on ECCV HotOrNot dataset. To the best of our knowledge, CRNet achieves the best performance.

Facial Attributes Analysis

- Hierarchical Multi-Task Network for Race, Gender and Facial Attractiveness Recognition

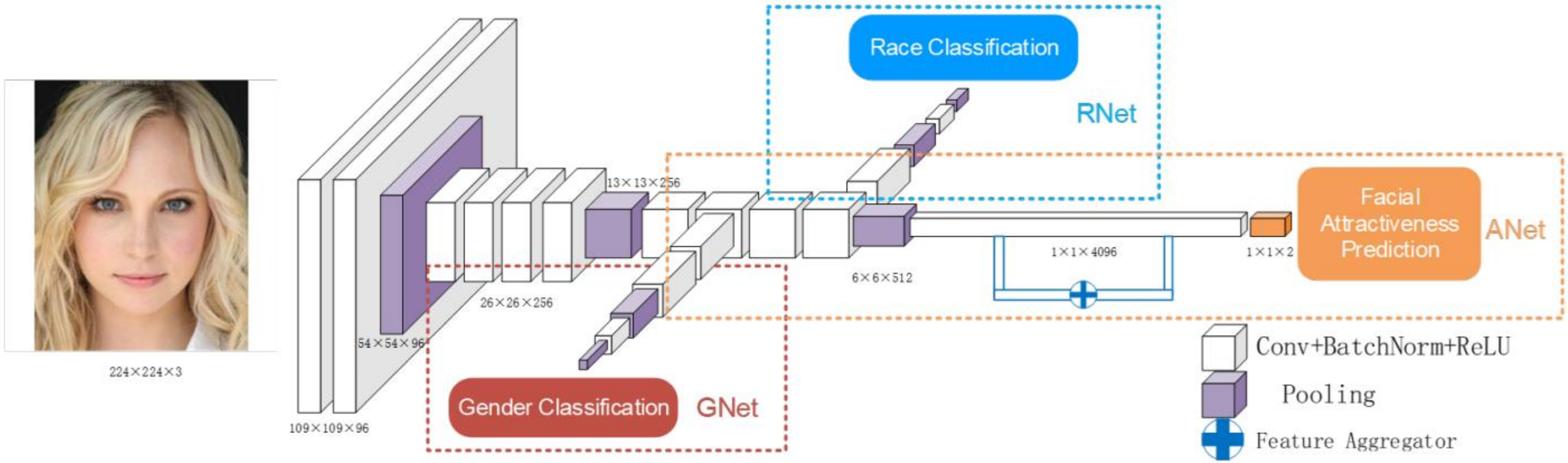


Fig. 1: Overall architecture of HMT-Net. RNet and GNet are used to recognize the race and gender, respectively. ANet is utilized to predict the facial attractiveness score. Lower layers can are shared among three sub-networks (GNet, RNet and ANet). All the layers are fully convolutional, and all three branched layers are trained jointly.

Facial Attributes Analysis

- Hierarchical Multi-Task Network for Race, Gender and Facial Attractiveness Recognition

$$Loss_g = -g log(\hat{g}) - (1-g) log(1-\hat{g})$$

$$Loss_r = -\sum_i r_i log(\hat{r}_i)$$

$$\boxed{Loss_a = \begin{cases} \sum_i log(cosh(a_i - \hat{a}_i)) & if |a_i - \hat{a}_i| \leq \delta \\ \sum_i \delta |a_i - \hat{a}_i| - \frac{1}{2}\delta^2 & otherwise \end{cases}}$$

$$Loss_{all} = \sum_{t \in \{g,r,a\}} \alpha_t Loss_t$$

$$f_{avg} = \frac{1}{\mathcal{C}} \sum_{i=1}^{\mathcal{C}} fm_i, \quad fm_i, f_{avg} \in \mathbb{R}^{w \times h \times c} \qquad (1)$$

$$f_{concat} = fm_1 \otimes \cdots \otimes fm_{\mathcal{C}}, \quad f_{concat} \in \mathbb{R}^{w \times h \times c \times \mathcal{C}} \qquad (2)$$

Facial Attributes Analysis

- Hierarchical Multi-Task Network for Race, Gender and Facial Attractiveness Recognition

Table 2: Performance comparison with other methods.

Model	MAE	RMSE	PC
AlexNet [12, 10]	0.2938	0.3819	0.8298
ResNet-18 [13, 10]	0.2818	0.3703	0.8513
ResNeXt-50 [14, 10]	0.2518	0.3325	0.8777
CRNet [21]	0.2835	0.3677	0.8558
HMT-Net (Ours)	**0.2501**	**0.3263**	**0.8783**

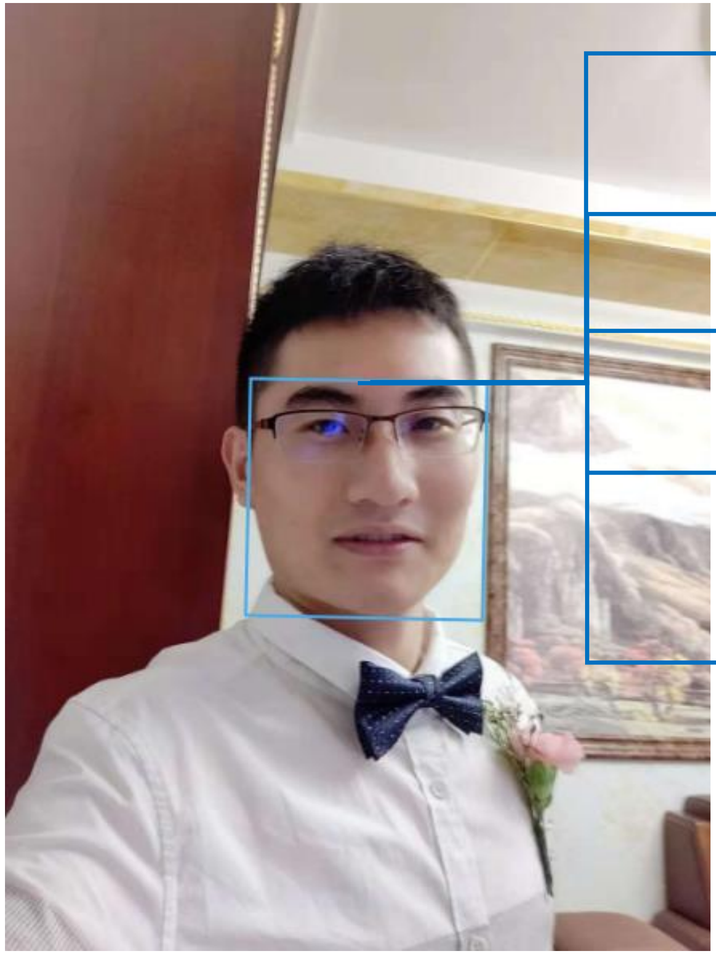
Table 3: We compare the performance on three tasks with or without jointly training, respectively.

With Joint Training			Without Joint Training		
Acc_r	Acc_g	PC	Acc_r	Acc_g	PC
99.26%	**98.16%**	**0.8783**	98.62%	97.56%	0.8616

Table 7: Comparison with other state-of-the-art models on [11]. PC is used as the performance metric as defined in [11].

Methods	PC
Combined Features+Gaussian Reg [11]	0.6482
CNN-based [11]	0.8187
Liu et al. [23]	0.6938
KFME [24]	0.7988
RegionScatNet [5]	0.83
PI-CNN [6]	0.87
CRNet [25]	0.8723
Transferred HMT-Net (Ours)	**0.8977**

Facial Attributes Analysis



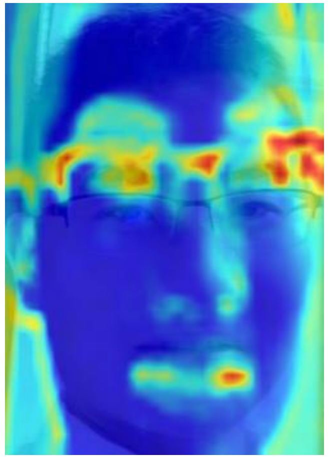
Beauty: 3.6

Gender: Male

Race: Asian

Expression: Neutral

Age: 25



Facial Attributes Analysis

- Multi-Task Tree Convolutional Neural Network for Facial Expression Recognition and Face Analysis

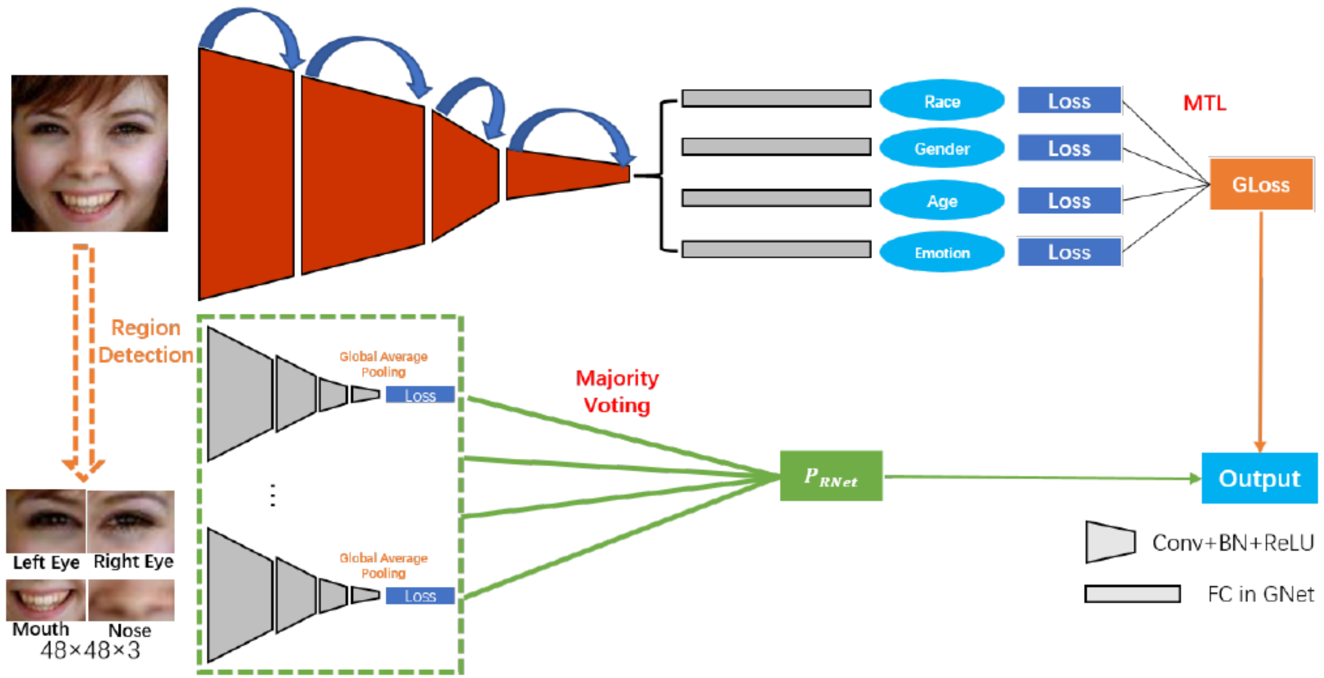


Figure 1: Overall architecture of TreeCNN. The facial part regions (left eye, right eye, nose and mouth) are detected and fed into a light-weighted *Region Network (RNet)* to obtain region information. The whole face is fed into a multi-task *Global Network (GNet)* to obtain global information. *GNet* follows an multi-task learning fashion, which indicates that *GNet* can perform FER and three additional face related recognition tasks (age estimation, gender recognition and race recognition) simultaneously. *RNet* follows an ensemble fashion, and the output is decided by majority voting.

Facial Attributes Analysis

- Multi-Task Tree Convolutional Neural Network for Facial Expression Recognition and Face Analysis

Multi-Task Learning

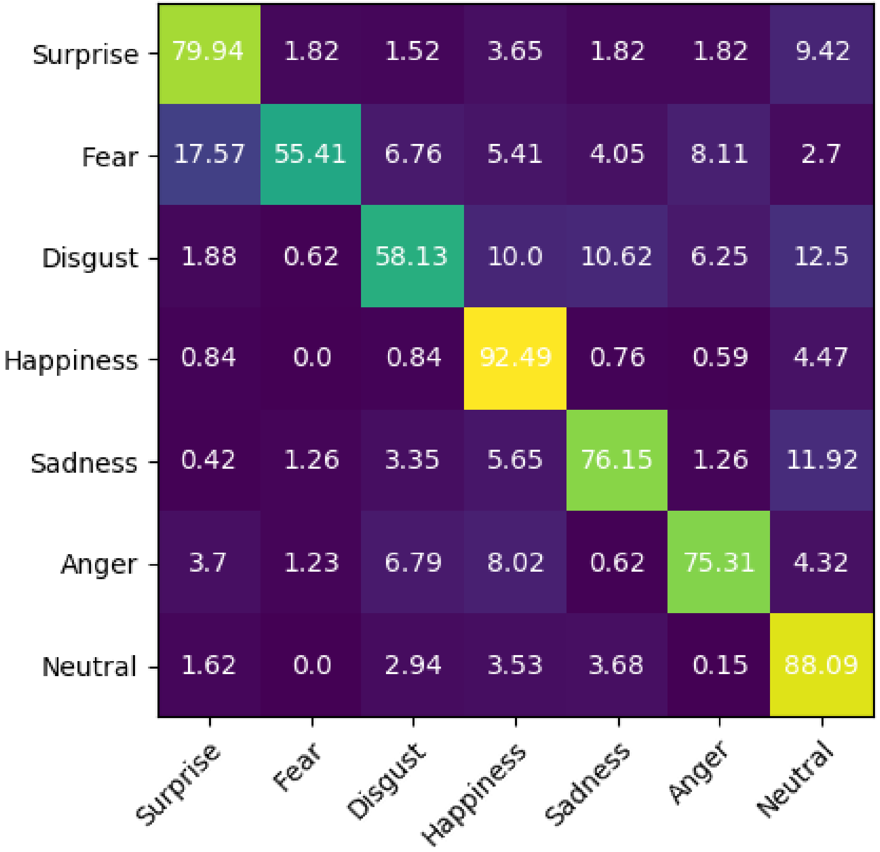
$$(\Theta_s^*, \Theta_{t_i}^*) = \underset{\Theta_s, \Theta_{t_i}}{argmin}\, \lambda_i \mathcal{L}_i(\Theta_s, \Theta_{t_i}; I) + \sum_{j \neq i}^{n} \lambda_j \mathcal{L}_j(\Theta_s, \Theta_{t_i}; I)$$

$$\mathcal{L}_{mt} = \sum_{i=1}^{N} \alpha_i Loss_i$$

$$Loss_i = -\sum_{c=1}^{M} y_c log \tilde{y}_c$$

Part Information Embedding

$$R(x) = c_{\underset{j}{argmax} \sum_{i=1}^{T} r_i^j(x)}$$



Facial Attributes Analysis

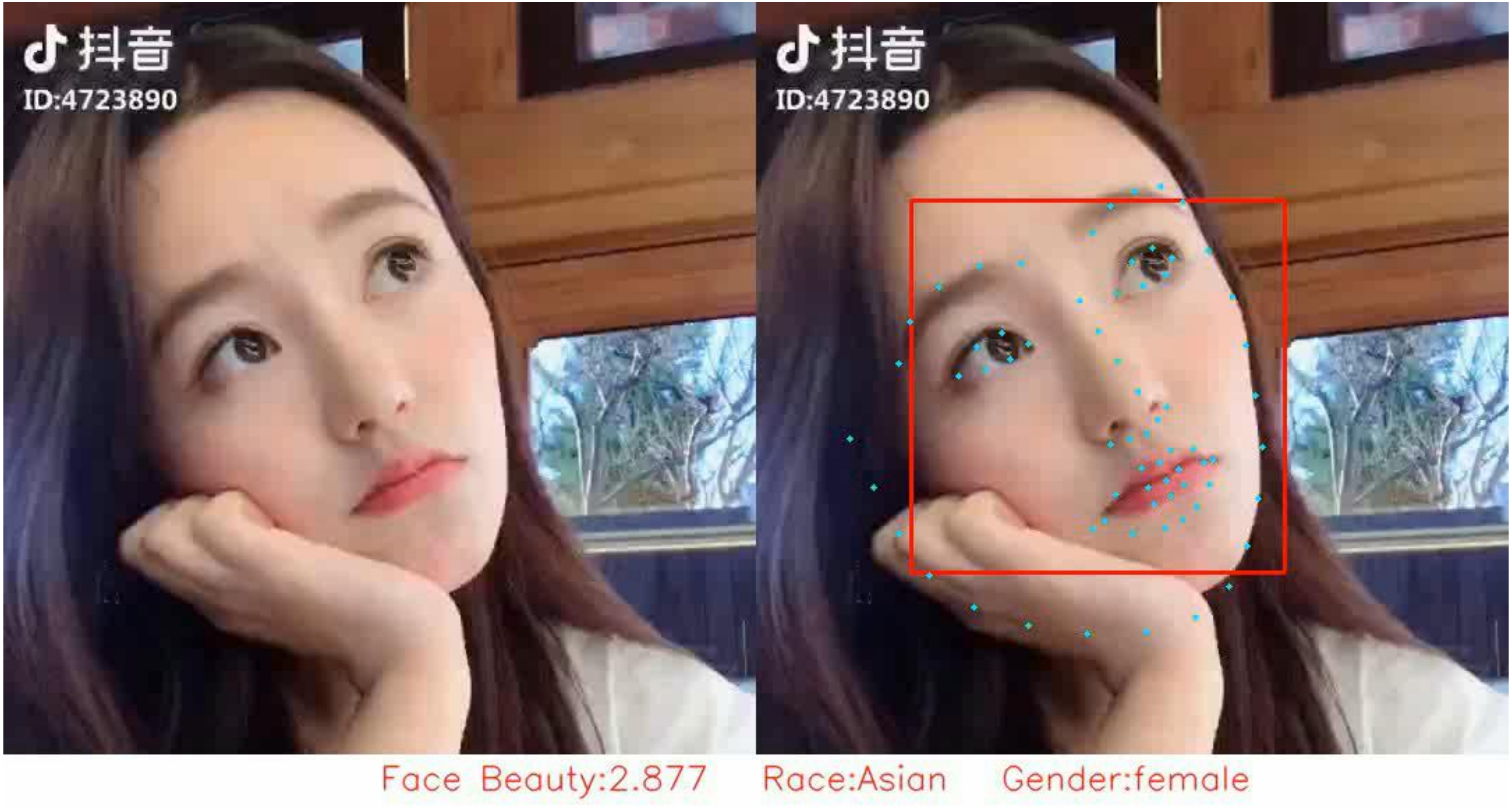
- Multi-Task Tree Convolutional Neural Network for Facial Expression Recognition and Face Analysis

Table 2. Performance comparison on RAF-DB [2] basic expressions with other state-of-the-art models and commercial APIs. TreeCNN outperforms other methods and achieves state-of-the-art performance.

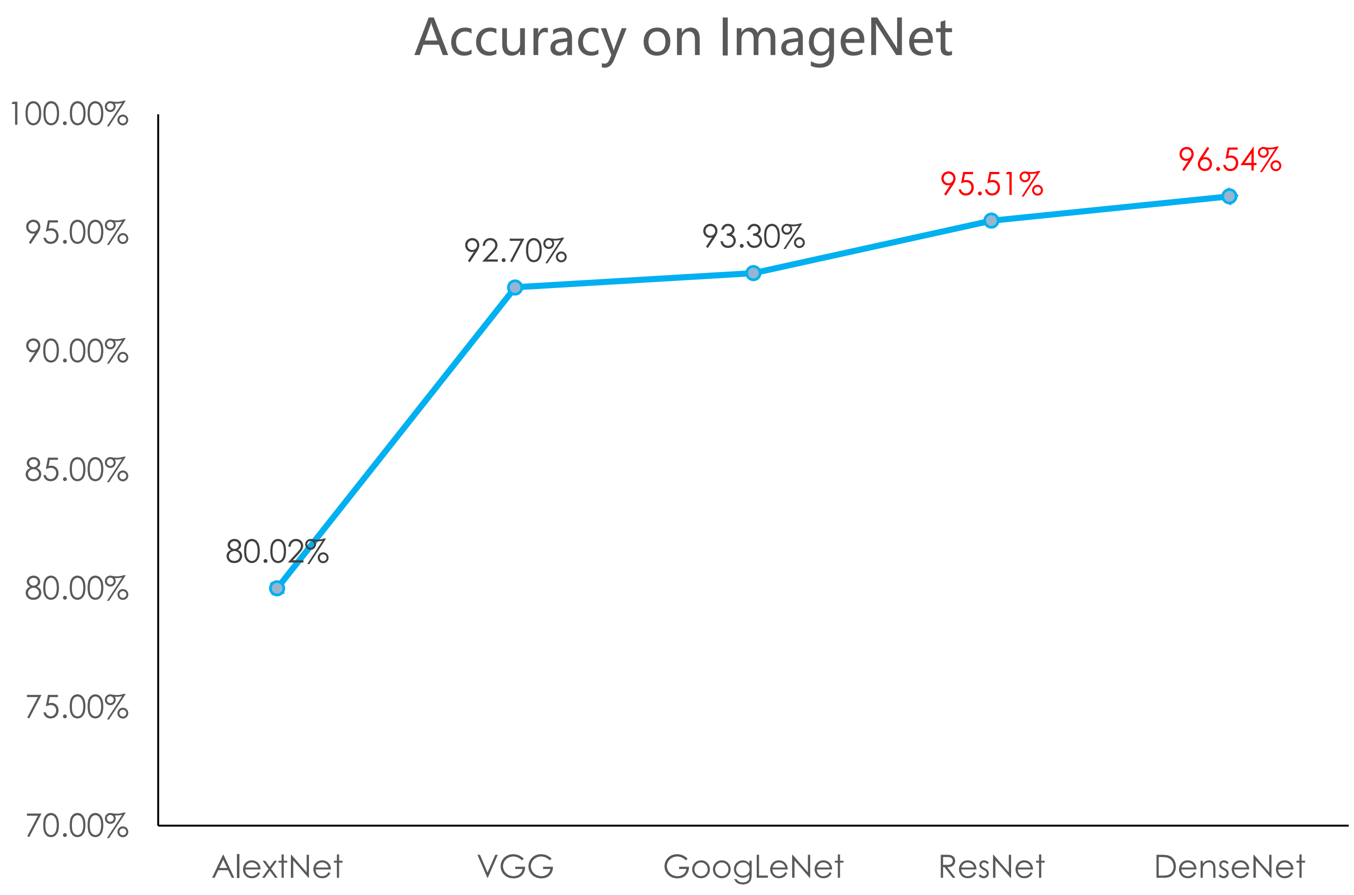
Methods	Anger	Disgust	Fear	Happiness	Sadness	Surprise	Neutral	**Average**
Face++ API [3]	48.89	17.24	24.19	77.39	45.69	30.14	55.16	42.66
baseline VGG [12]	68.52	27.50	35.13	85.32	64.85	66.32	59.88	58.22
baseDCNN [2]	70.99	52.50	50.00	92.91	77.82	79.64	83.09	72.42
DLP-CNN [2]	71.60	52.15	62.16	92.83	80.13	81.16	80.29	74.20
Wen et al. [30]	68.52	53.13	54.05	93.08	78.45	79.63	83.24	72.87
Kuo et al. [31]	74.47	67.57	46.88	82.28	57.95	84.57	59.12	67.55
MRE-CNN [32]	83.95	57.50	60.81	88.78	79.92	86.02	80.15	76.73
Kervadec et al. [33]	-	-	-	-	-	-	-	71.7
TreeCNN (Ours)	75.31	58.13	55.41	92.49	76.15	79.94	88.09	$\boxed{\textbf{78.49}}$

Our proposed TreeCNN ranks the 1st place compared with all prior state-of-the-arts.

02 Deep Learning for Face Analysis



03 Deep Learning for Fine-grained Visual Recognition



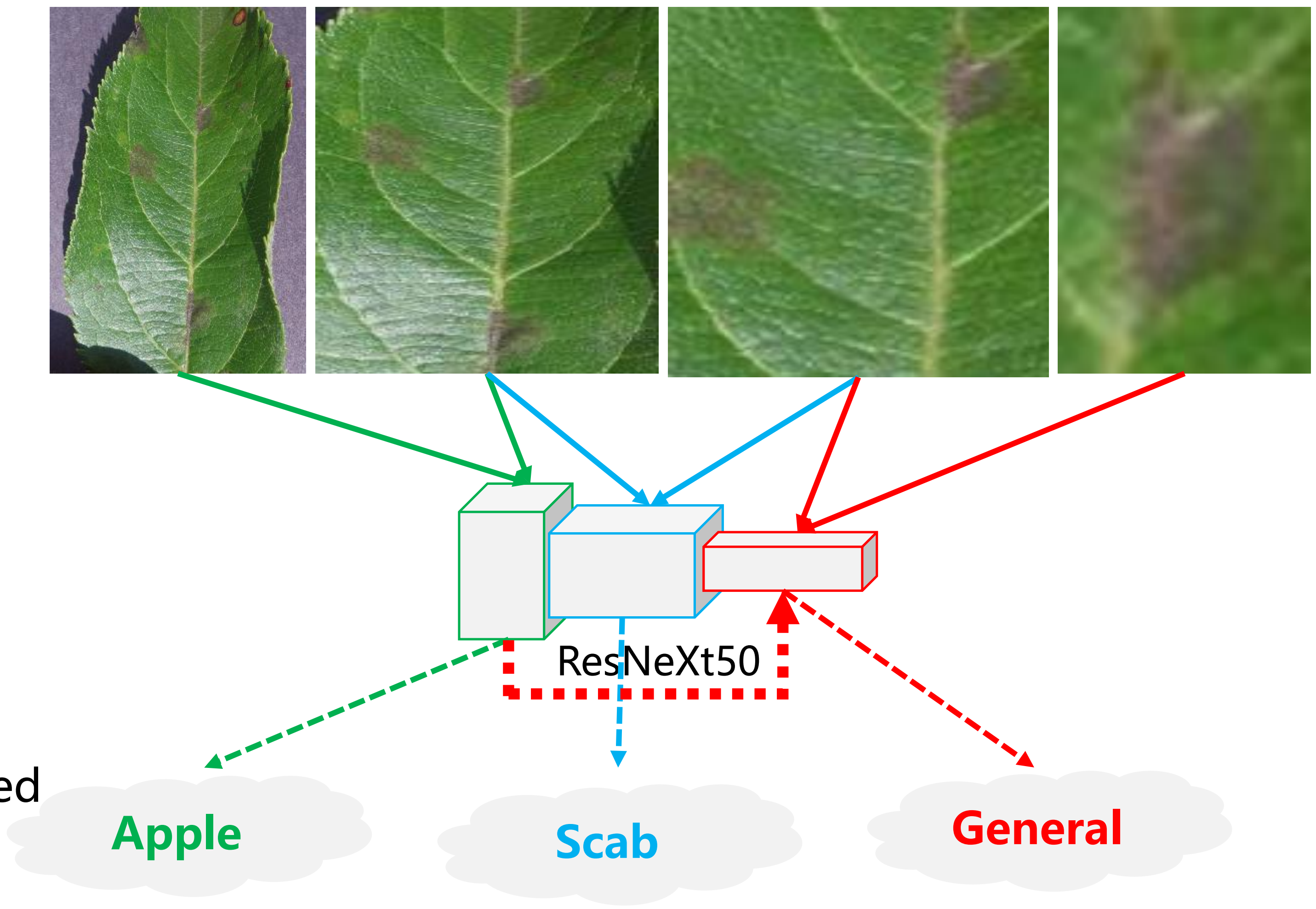
Does Deep Learning Really Surpass Human on Visual Recognition?

03 Deep Learning for Fine-grained Visual Recognition

How About These?

A Coarse-to-fine Method for **Fine-grained** Visual Recognition

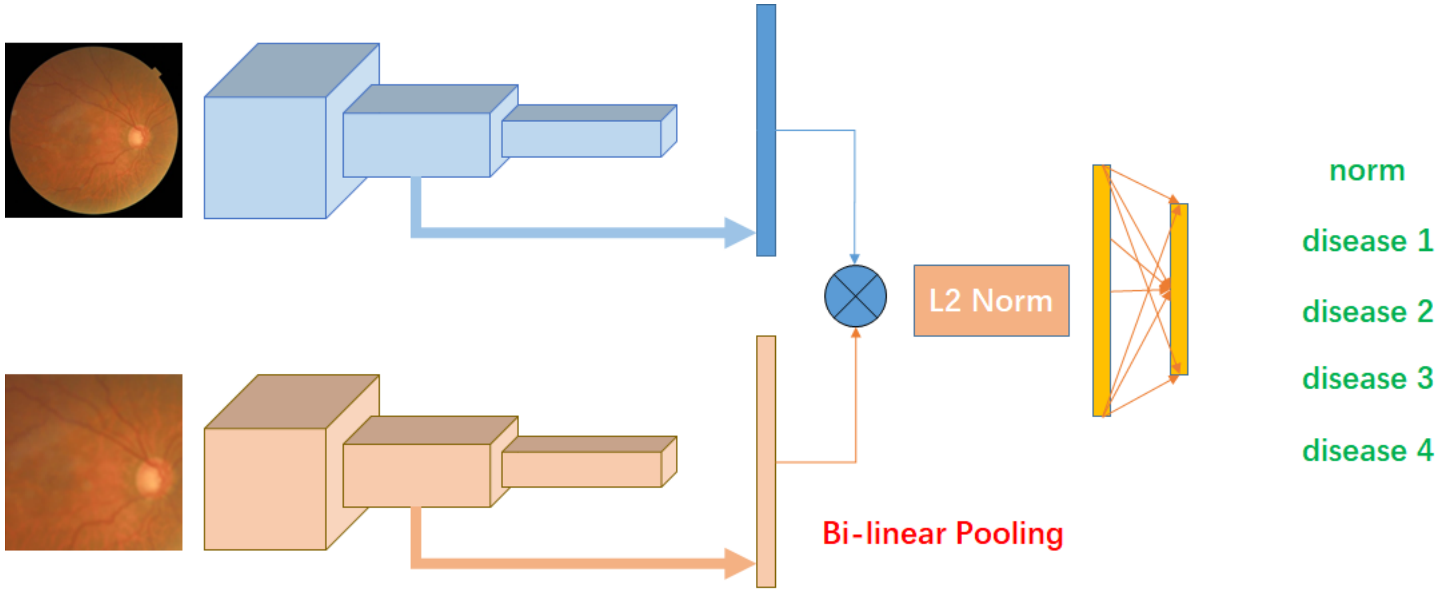
- **Data Imbalance**
  - Over Sampling
  - Mix-up
  - Weighted Softmax Loss
- **Multi-level Categorization**
  - Coarse-to-fine Classification
- **Fine-grained Feature Learning**
  - Zoom Data Augmentation
  - Feature Pyramid



Our proposed method achieves over 87% precision on the challenging 61 fine-grained classification task.

Applicable Deep Learning for Eye Disease Recognition

- Easy Ensemble
- Bilinear Pooling
- Transfer Learning



Only 82% precision…

04 XCloud: From Research to Production

MTCNN + FaceNet + L2 Distance + Django + MySQL

(Cooperate With Guangzhou Yingzi Technology)

XCloud

- Pure Python (Django + PyTorch)
- Better Network Architecture
  (30× Faster, 408ms/Per Image on PC)
- Bridge the Gap Between Research and Production
- Permanently Free and Open Source
- Current Partners (PKU, HKU)
- API has been called over 200,000 times

Web Data Mining

- Data-driven Approach for Quality Evaluation on Knowledge Sharing Platform

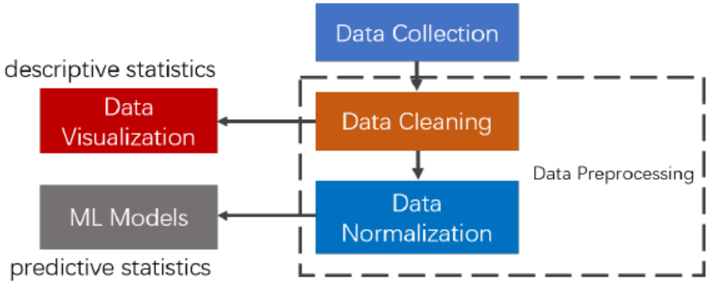


Fig. 1: The architecture of our data-driven method. The records are crawled from Zhihu Live official website and stored in MongoDB. Data preprocessing methods include cleaning and data normalization to make the dataset satisfy our target problem. We make detailed data analysis and predictive analysis.

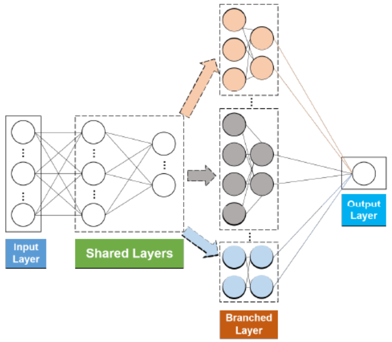


Fig. 2: Overall architecture of multi-branched deep neural network (MTB-DNN). It includes 4 parts: an input layer for receiving raw data; shared layers for general feature extraction through stacked layers and non-linear transformation; branched layers for specific feature extraction; and the output layer with one neuron. The output of the last shared layer is fed into different branches. These branches are trained jointly.

Table 7: Performance comparison with baseline regression models.

Regressor	MAE	RMSE
Ridged Regression	0.309 ± 0.01015554	0.41716 ± 0.015474592
Lasso Regression	0.35038 ± 0.016164065	0.46916 ± 0.032221778
KNN Regression	0.32328 ± 0.006829129	$0.43888 \pm \mathbf{0.006319968}$
SVR (RBF)	0.31022 ± 0.011957508	0.43322 ± 0.022196892
SVR (Linear)	$0.30134 \pm \mathbf{0.005944998}$	0.424 ± 0.016474374
SVR (Poly)	0.29974 ± 0.009122938	0.4208 ± 0.013073255
MLP	0.32024 ± 0.015835814	0.43496 ± 0.017316842
MTB-DNN	$\mathbf{0.29954} \pm 0.012644485$	$\mathbf{0.40114} \pm 0.011662461$

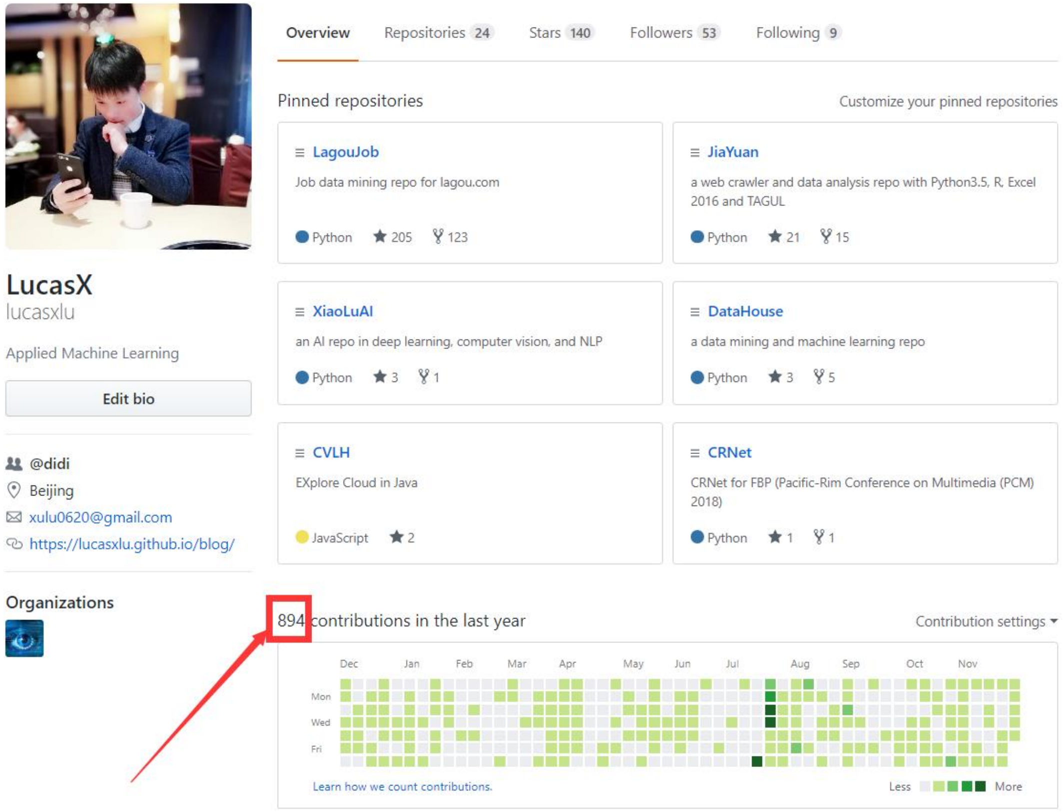
Experimental results between different regression algorithms. The architecture of MLP is 15-16-8-8-1, where each number represents the number of neurons in each layer. We try three kinds of kernels (RBF kernel, linear kernel, and poly kernel) with SVM regression. The best results are given in bold style.

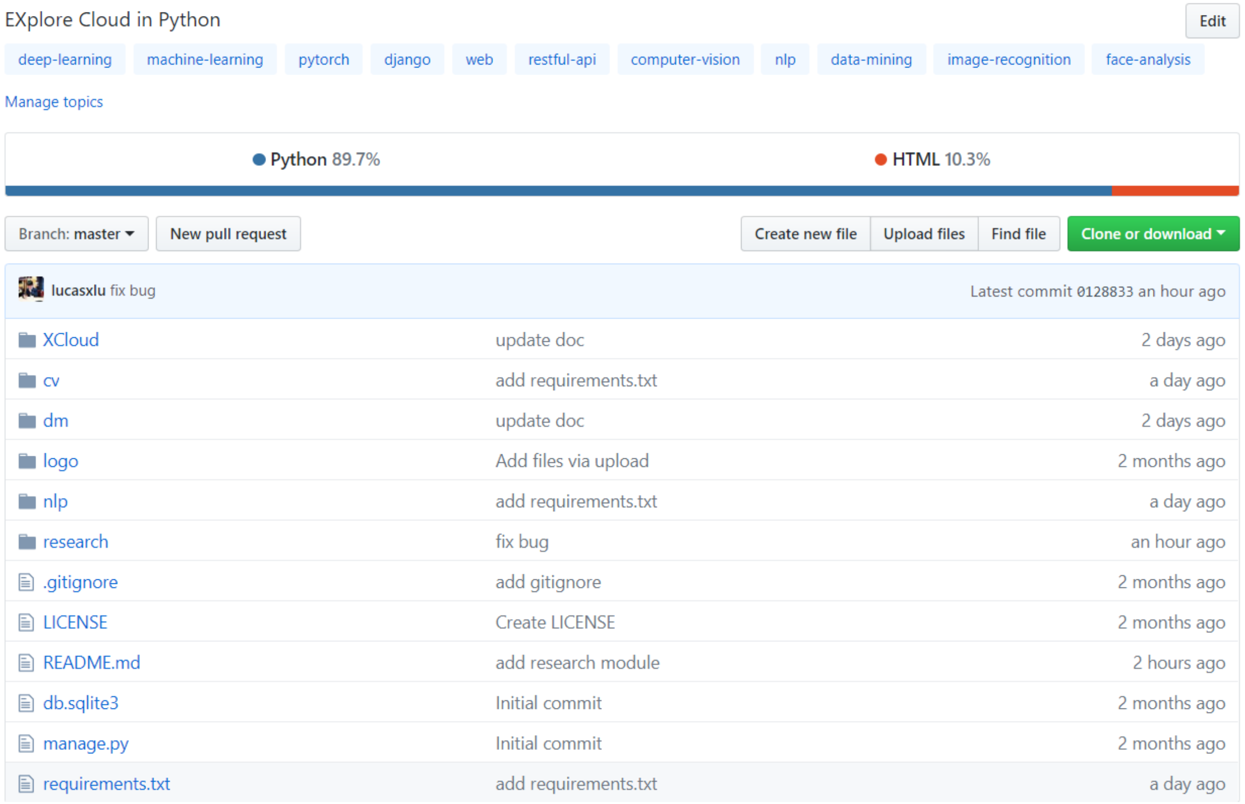
Open Source

https://github.com/lucasxlu/XCloud.git

Github/知乎: @LucasX

04 XCloud: From Research to Production





- **No** Software Copyrights
- **No** Additional Authorization
- Under **MIT** License

Thanks!

Speaker：Lu Xu

Supervisor: Jinhai Xiang

Research interests: Deep Learning/Computer Vision