Masked-face recognition based on Attention

- Group2: Darling's Honey Home

LENG Tianang	Model modifying & Literature Survey
LIU Yufei	Dataset Producing
KUANG Haohong	Model modifying & Literature Survey
WANG Rongying	Dataset Cleaning & Producing

Catalogue

By Leng Tianang

Background

COVID-19 breaking out all over the world

Existing systems fail to recognize us

By Leng Tianang

Literature Review

Main ideas:

1. Utilizing a generative adversarial network (GAN) to perform face completion so that the content under the mask could be recovered.
2. Adding CBAM or SE module into Resnet baseline, to focus on areas with more features.
3. Hybrid transformer with CNN (ICCV workshop)

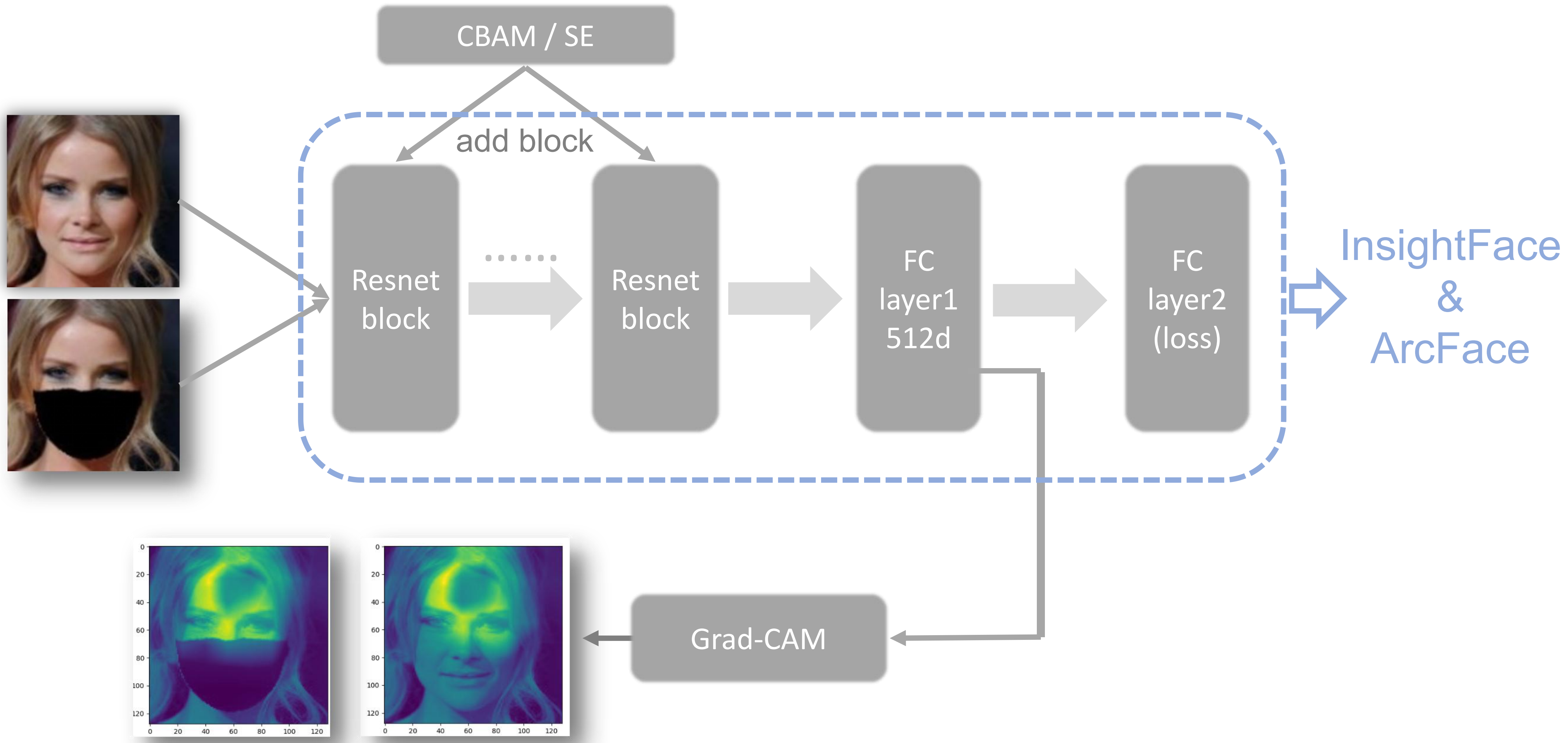
[1] Chenyu Li, Shiming Ge, Daichi Zhang, and Jia Li. Look through masks: Towards masked face recognition with deocclusion distillation. ACM Multimedia, 2020..

[2] Li, Yande, et al. "Cropping and attention based approach for masked face recognition." Applied Intelligence 51.5 (2021): 3012-3025.

[3] Chang, Wei-Yi, Ming-Ying Tsai, and Shih-Chieh Lo. "ResSaNet: A Hybrid Backbone of Residual Block and Self-Attention Module for Masked Face Recognition." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021.

Project structure

By Leng Tianang



By Leng Tianang

Why we choose Arcface to be baseline

Arcface outperforms other methods greatly on both identification and verification task.

Methods	Id (%)	Ver (%)
Softmax [18]	54.85	65.92
Contrastive Loss[18, 32]	65.21	78.86
Triplet [18, 29]	64.79	78.32
Center Loss[38]	65.49	80.14
SphereFace [18]	72.729	85.561
CosFace [37]	77.11	89.88
AM-Softmax [35]	72.47	84.44
SphereFace+ [17]	73.03	-
CASIA, R50, ArcFace	77.50	92.34
CASIA, R50, ArcFace, R	91.75	93.69
FaceNet [29]	70.49	86.47
CosFace [37]	82.72	96.65
MS1MV2, R100, ArcFace	81.03	96.98
MS1MV2, R100, CosFace	80.56	96.56
MS1MV2, R100, ArcFace, R	98.35	98.48
MS1MV2, R100, CosFace, R	97.91	97.91

Deng, Jiankang, et al. "Arcface: Additive angular margin loss for deep face recognition." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.

Why Arcface wins ?

By Leng Tianang

The most important thing is the loss function

$$L_1 = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{W_{y_i}^T x_i + b_{y_i}}}{\sum_{j=1}^{n} e^{W_j^T x_i + b_j}} \rightarrow L_3 = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s(\cos(\theta_{y_i}+m))}}{e^{s(\cos(\theta_{y_i}+m))} + \sum_{j=1, j\neq y_i}^{n} e^{s\cos\theta_j}}$$

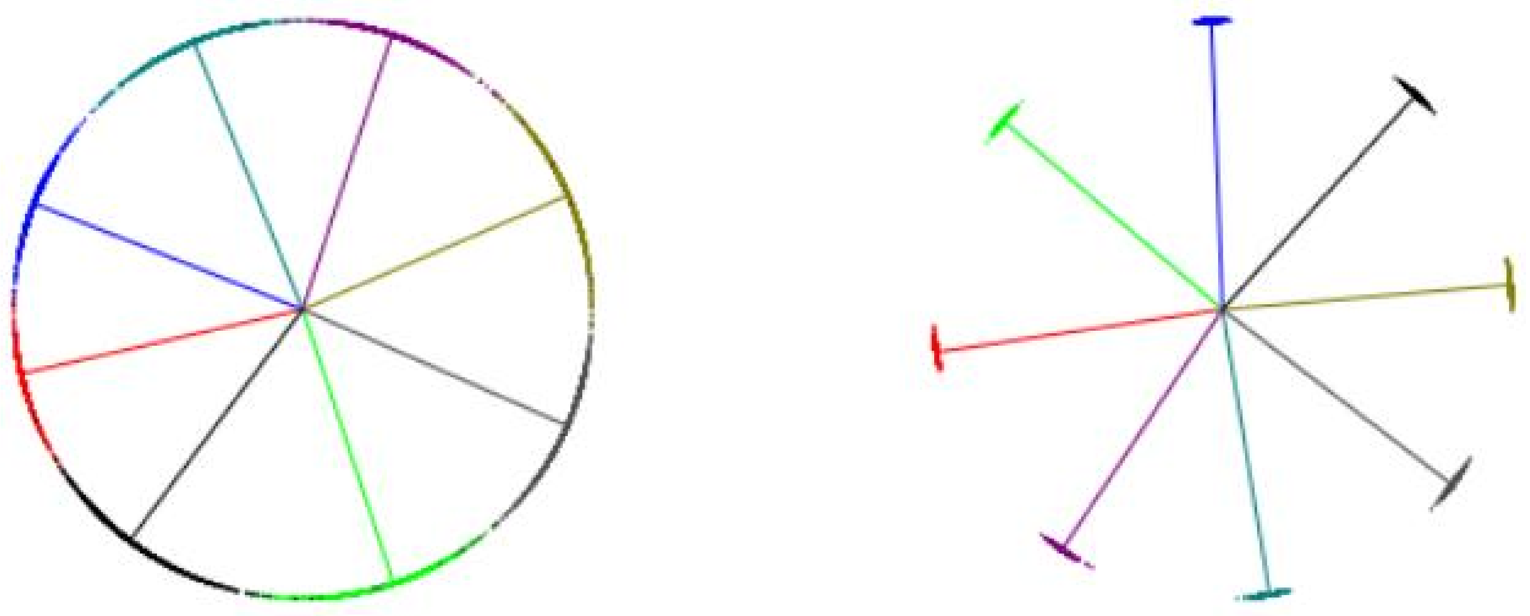
Softmax-loss function

Additive Angular Margin Loss

Deng, Jiankang, et al. "Arcface: Additive angular margin loss for deep face recognition." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.

By Leng Tianang

Comparison between two loss functions

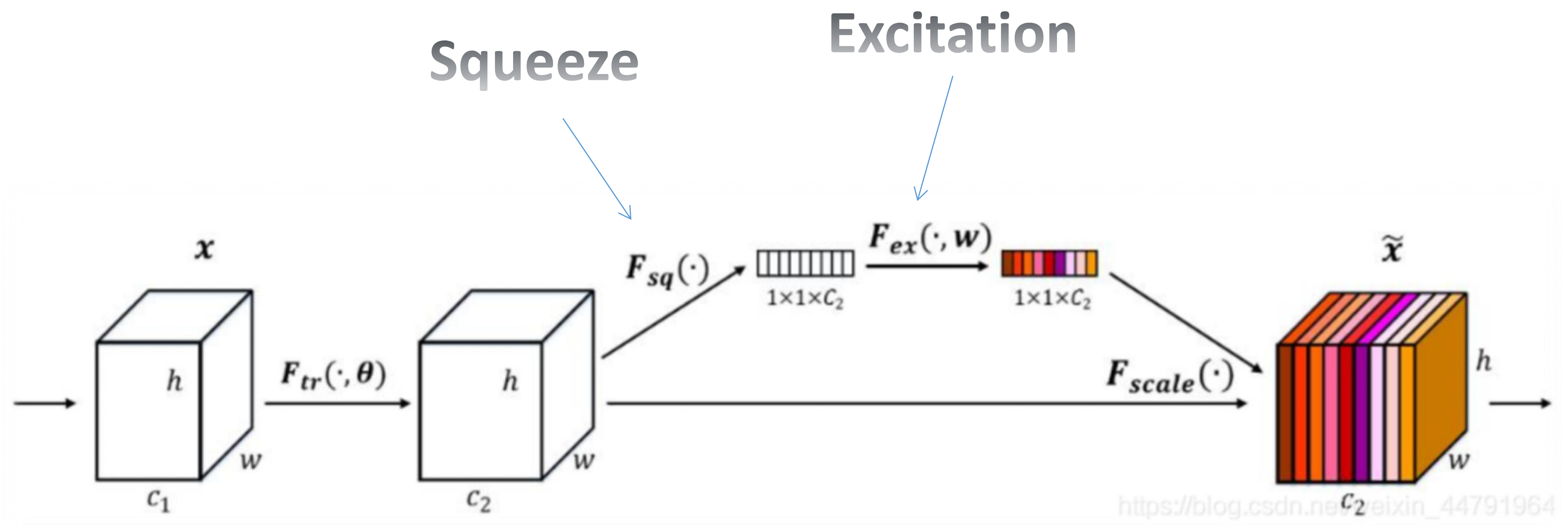


(a) Softmax (b) ArcFace

Deng, Jiankang, et al. "Arcface: Additive angular margin loss for deep face recognition." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.

By LIU YUFEI

Attention Machanism ——— SE



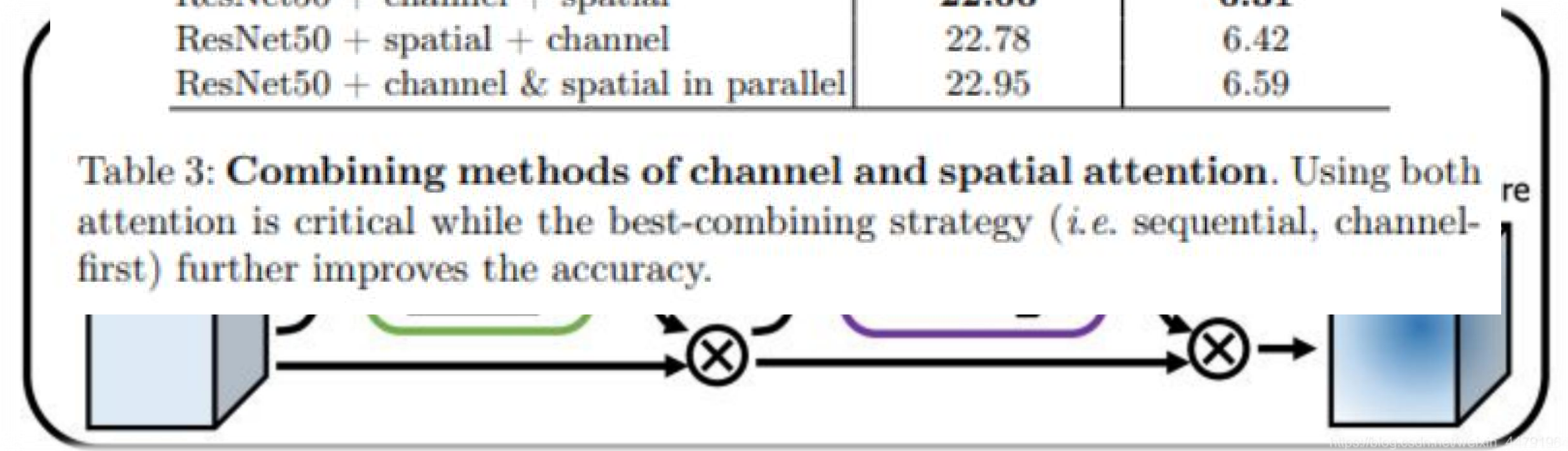
Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7132-7141.

By LIU YUFEI

Attention Machanism —— CBAM

Description	Top-1 Error(%)	Top-5 Error(%)
ResNet50 + channel (SE [28])	23.14	6.70
ResNet50 + channel + spatial	**22.66**	**6.31**
ResNet50 + spatial + channel	22.78	6.42
ResNet50 + channel & spatial in parallel	22.95	6.59

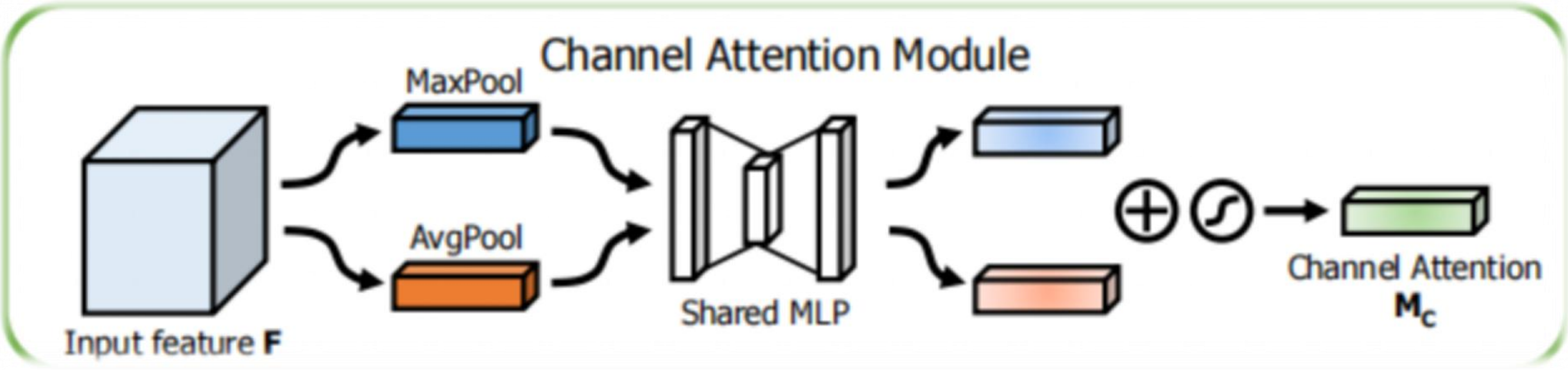
Table 3: **Combining methods of channel and spatial attention**. Using both attention is critical while the best-combining strategy (*i.e.* sequential, channel-first) further improves the accuracy.



Woo S, Park J, Lee J Y, et al. Cbam: Convolutional block attention module[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 3-19.

By LIU YUFEI

Attention Machanism ——— CBAM

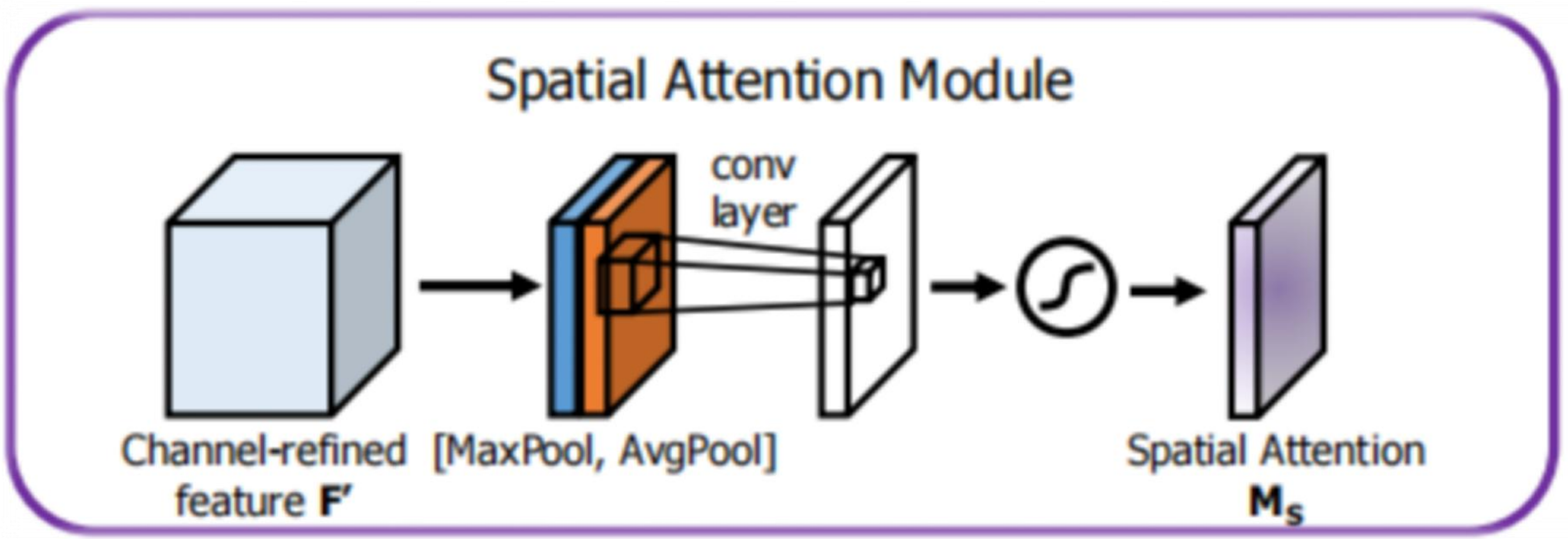


$$\begin{aligned}\mathbf{M_c}(\mathbf{F}) &= \sigma(MLP(\text{ AvgPool }(\mathbf{F})) + MLP(\text{MaxPool}(\mathbf{F}))) \\ &= \sigma\left(\mathbf{W_1}\left(\mathbf{W_0}\left(\mathbf{F}^{\mathbf{c}}_{\mathbf{avg}}\right)\right) + \mathbf{W_1}\left(\mathbf{W_0}\left(\mathbf{F}^{\mathbf{c}}_{\mathbf{max}}\right)\right)\right),\end{aligned}$$

Woo S, Park J, Lee J Y, et al. Cbam: Convolutional block attention module[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 3-19.

By LIU YUFEI

Attention Machanism —— CBAM



$$
\begin{aligned}
\mathbf{M}_{\mathbf{s}}(\mathbf{F}) &= \sigma\left(f^{7\times 7}([\mathrm{AvgPool}(\mathbf{F}); \mathrm{MaxPool}(\mathbf{F})])\right) \\
&= \sigma\left(f^{7\times 7}\left([\mathbf{F}^{\mathbf{s}}_{\mathbf{avg}}; \mathbf{F}^{\mathbf{s}}_{\mathbf{max}}]\right)\right)
\end{aligned}
$$

Woo S, Park J, Lee J Y, et al. Cbam: Convolutional block attention module[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 3-19.

By LIU YUFEI

Attention Machanism ——CBAM

```
class ChannelAttention(nn.Module):
    def __init__(self, in_planes, ratio=16):
        super(ChannelAttention, self).__init__()
        self.avg_pool = nn.AdaptiveAvgPool2d(1)
        self.max_pool = nn.AdaptiveMaxPool2d(1)

        self.fc = nn.Sequential(nn.Conv2d(in_planes, in_planes // 16, 1, bias=False),
                                nn.ReLU(),
                                nn.Conv2d(in_planes // 16, in_planes, 1, bias=False))
        self.sigmoid = nn.Sigmoid()

    def forward(self, x):
        avg_out = self.fc(self.avg_pool(x))
        max_out = self.fc(self.max_pool(x))
        out = avg_out + max_out
        return self.sigmoid(out)
```

```
class SpatialAttention(nn.Module):
    def __init__(self, kernel_size=7):
        super(SpatialAttention, self).__init__()

        self.conv1 = nn.Conv2d(2, 1, kernel_size, padding=kernel_size // 2, bias=False)
        self.sigmoid = nn.Sigmoid()

    def forward(self, x):
        avg_out = torch.mean(x, dim=1, keepdim=True)
        max_out, _ = torch.max(x, dim=1, keepdim=True)
        x = torch.cat([avg_out, max_out], dim=1)
        x = self.conv1(x)
        return self.sigmoid(x)
```

Attention Machanism —— CBAM

By LIU YUFEI

```python
class IRBlock(nn.Module):
    expansion = 1

    def __init__(self, inplanes, planes, stride=1, downsample=None, use_se=True):
        super(IRBlock, self).__init__()
        self.bn0 = nn.BatchNorm2d(inplanes)
        self.conv1 = conv3x3(inplanes, inplanes)
        self.bn1 = nn.BatchNorm2d(inplanes)
        self.prelu = nn.PReLU()
        self.conv2 = conv3x3(inplanes, planes, stride)
        self.bn2 = nn.BatchNorm2d(planes)
        self.downsample = downsample
        self.stride = stride
        self.use_se = use_se
        if self.use_se:
            self.se = SEBlock(planes)

    def forward(self, x):
        residual = x
        out = self.bn0(x)
        out = self.conv1(out)
        out = self.bn1(out)
        out = self.prelu(out)

        out = self.conv2(out)
        out = self.bn2(out)
        if self.use_se:
            out = self.se(out)

        if self.downsample is not None:
            residual = self.downsample(x)

        out += residual
        out = self.prelu(out)

        return out
```

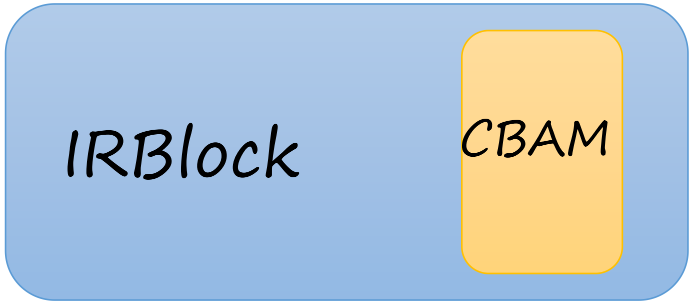
```python
class IRBlock_cbam(nn.Module):
    expansion = 1

    def __init__(self, inplanes, planes, stride=1, downsample=None, use_se=False):
        super(IRBlock_cbam, self).__init__()
        self.bn0 = nn.BatchNorm2d(inplanes)
        self.conv1 = conv3x3(inplanes, inplanes)
        self.bn1 = nn.BatchNorm2d(inplanes)
        self.prelu = nn.PReLU()
        self.conv2 = conv3x3(inplanes, planes, stride)
        self.bn2 = nn.BatchNorm2d(planes)
        self.downsample = downsample
        self.stride = stride
        self.ca = ChannelAttention(planes)
        self.sa = SpatialAttention()

    def forward(self, x):
        residual = x
        out = self.bn0(x)
        out = self.conv1(out)
        out = self.bn1(out)
        out = self.prelu(out)
        out = self.conv2(out)
        out = self.bn2(out)
        out = self.ca(out) * out
        out = self.sa(out) * out
        if self.downsample is not None:
            residual = self.downsample(x)

        out += residual
        out = self.prelu(out)

        return out
```



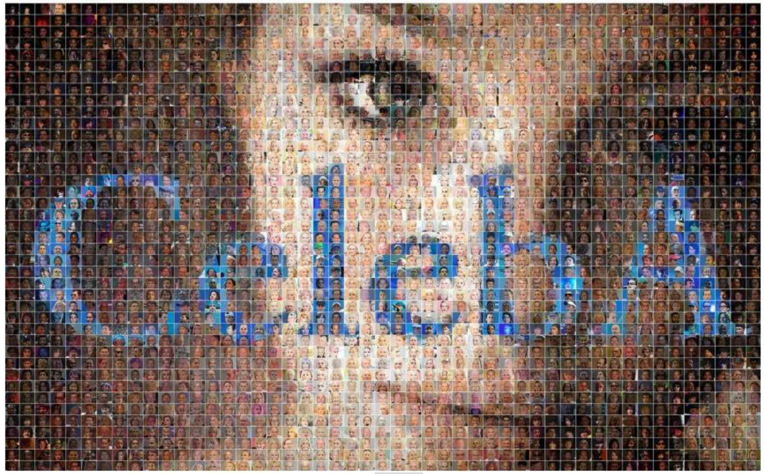
Our dataset

Kuang Haohong

About 50000 images

7/10 → Train set

3/10 → Test set

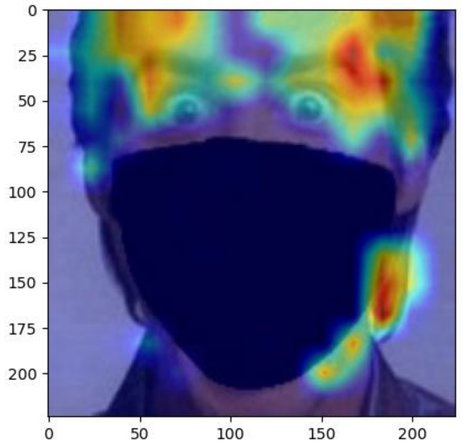


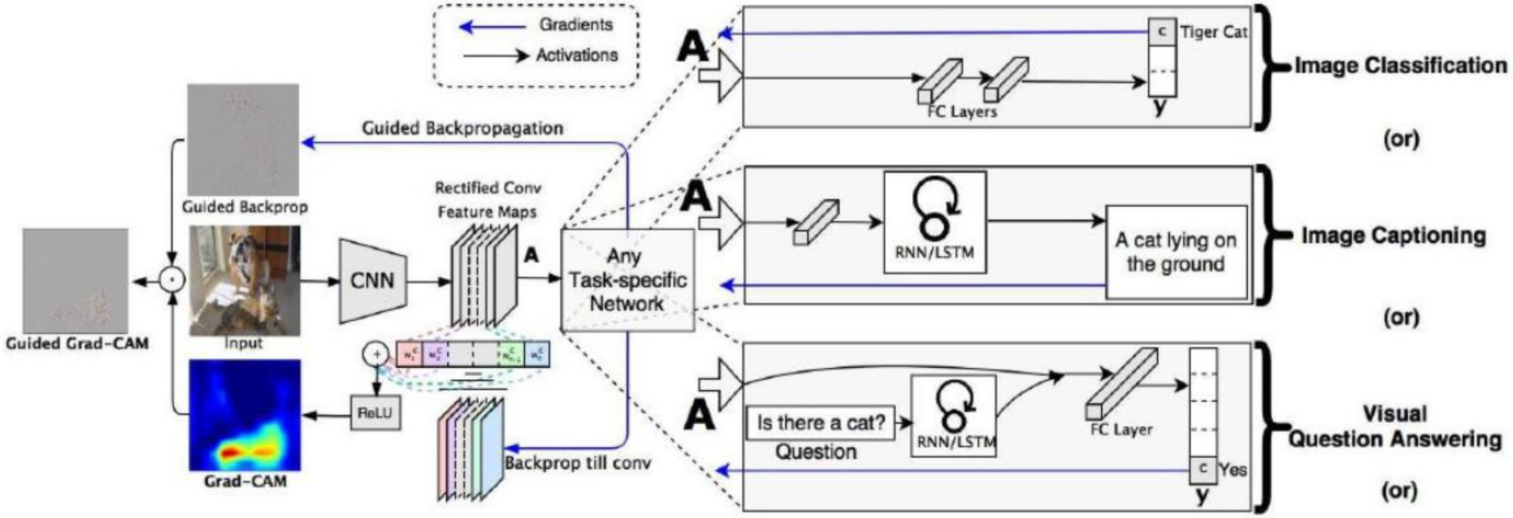
Face_Mask_Add:

Black mask

Tool: Grad-CAM

To track the effect of training, we use Grad-CAM to visualize the attention of the model

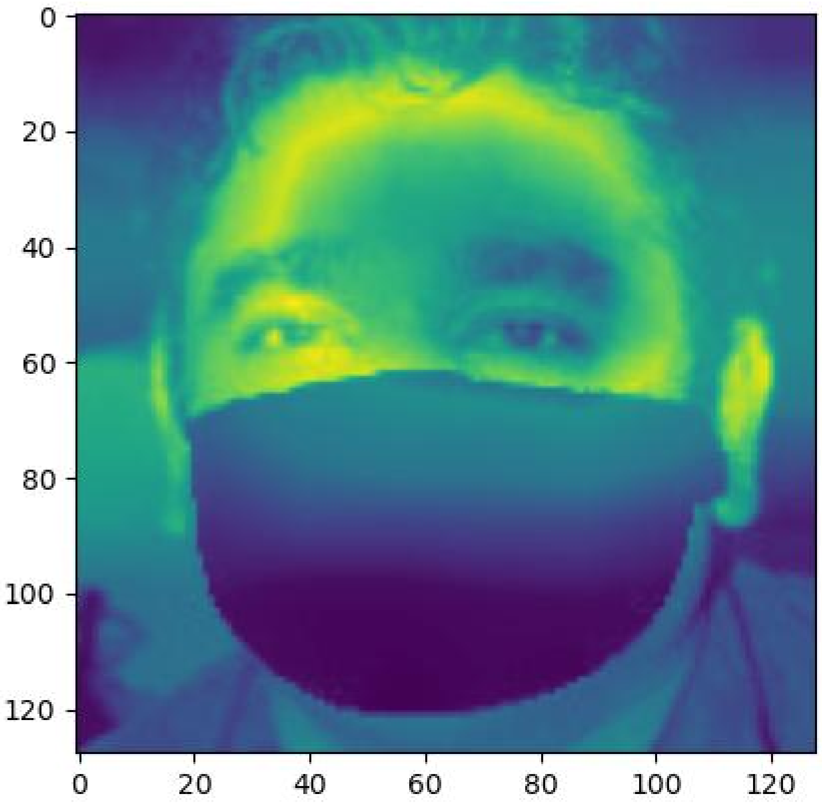




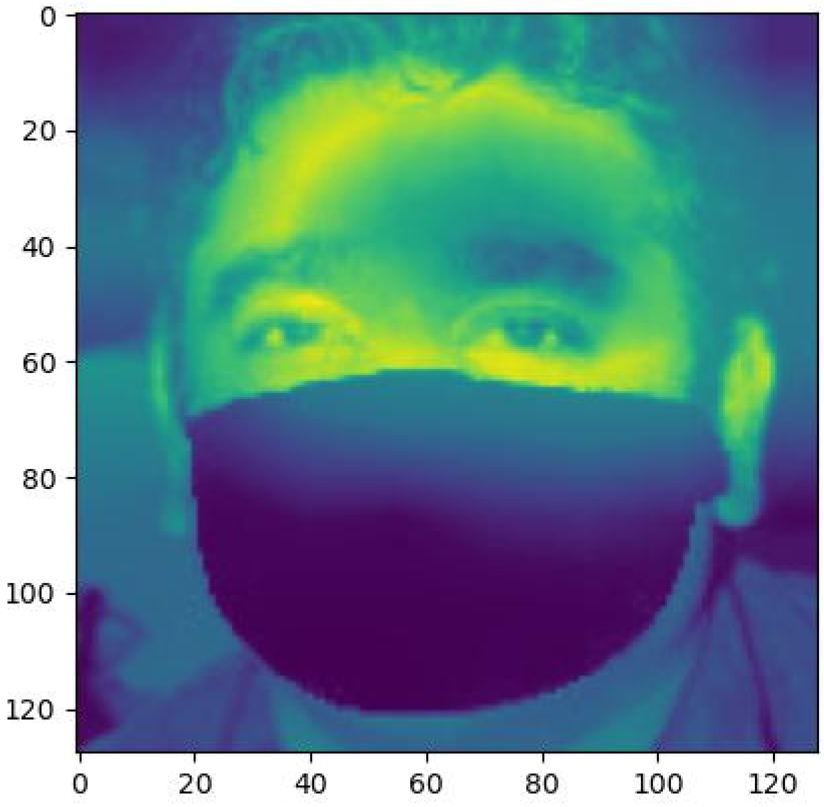
Selvaraju, Ramprasaath R., et al. "Grad-cam: Visual explanations from deep networks via gradient-based localization." Proceedings of the IEEE international conference on computer vision. 2017.

By Leng Tianang

Grad-CAM output analyzation



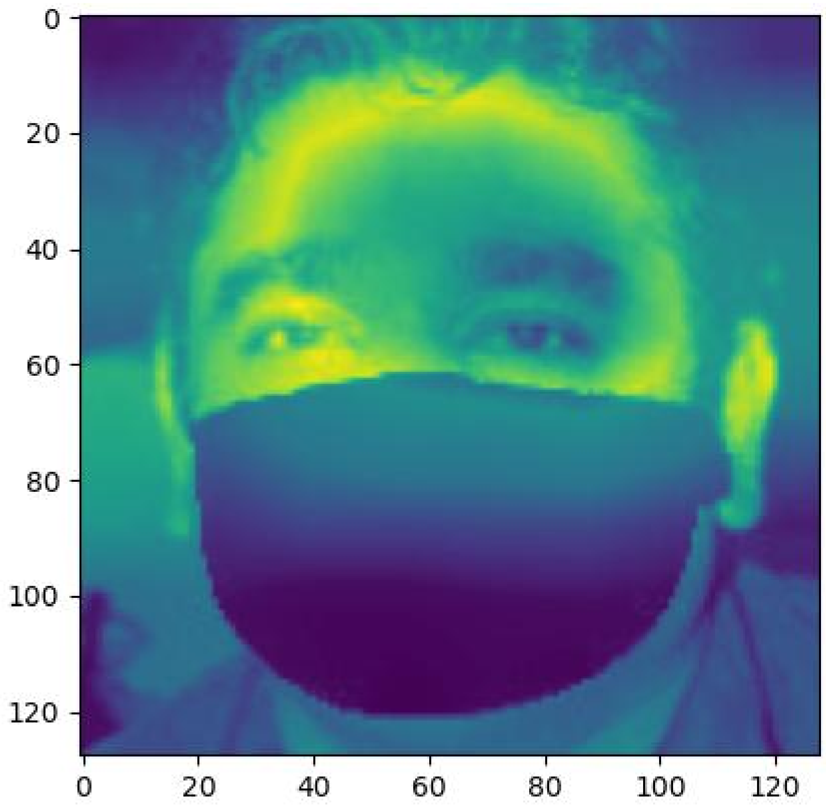
Without CBAM



With CBAM

By Leng Tianang

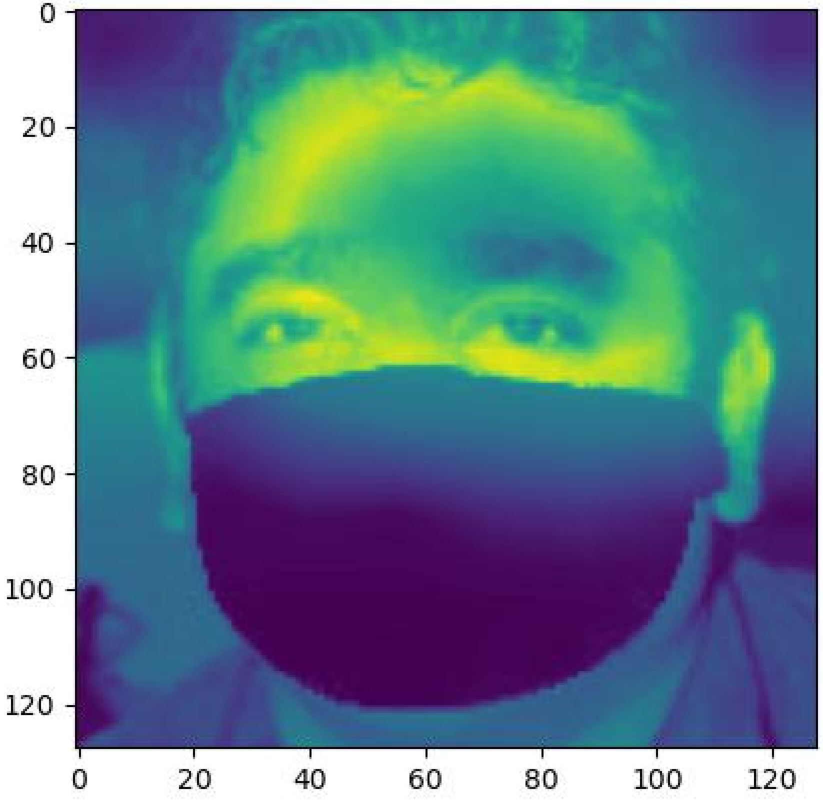
Grad-CAM output analyzation



Without CBAM

Grad-CAM output analyzation

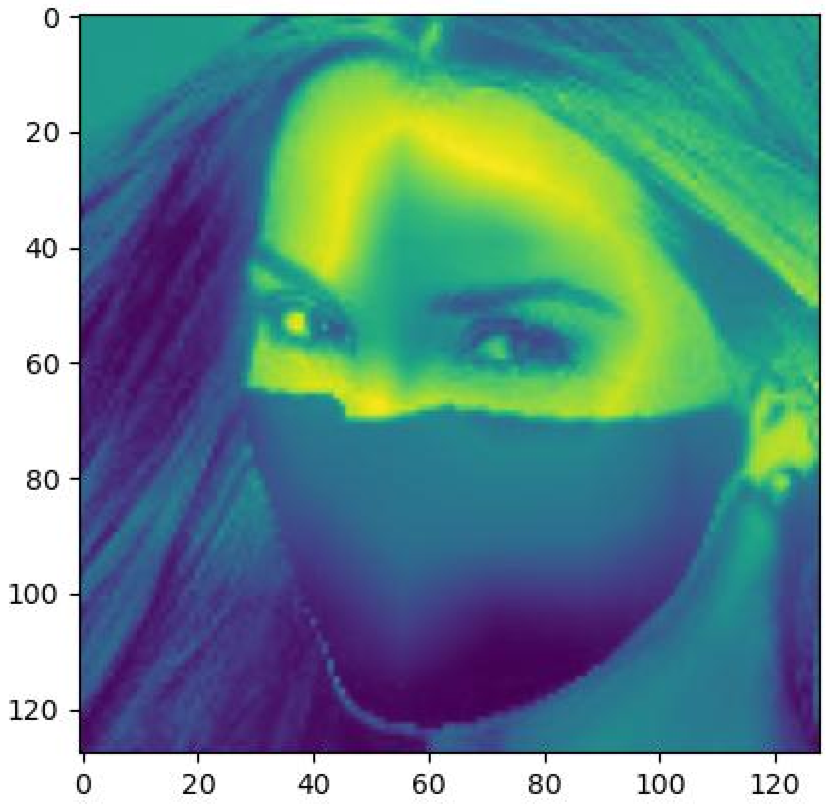
By Leng Tianang

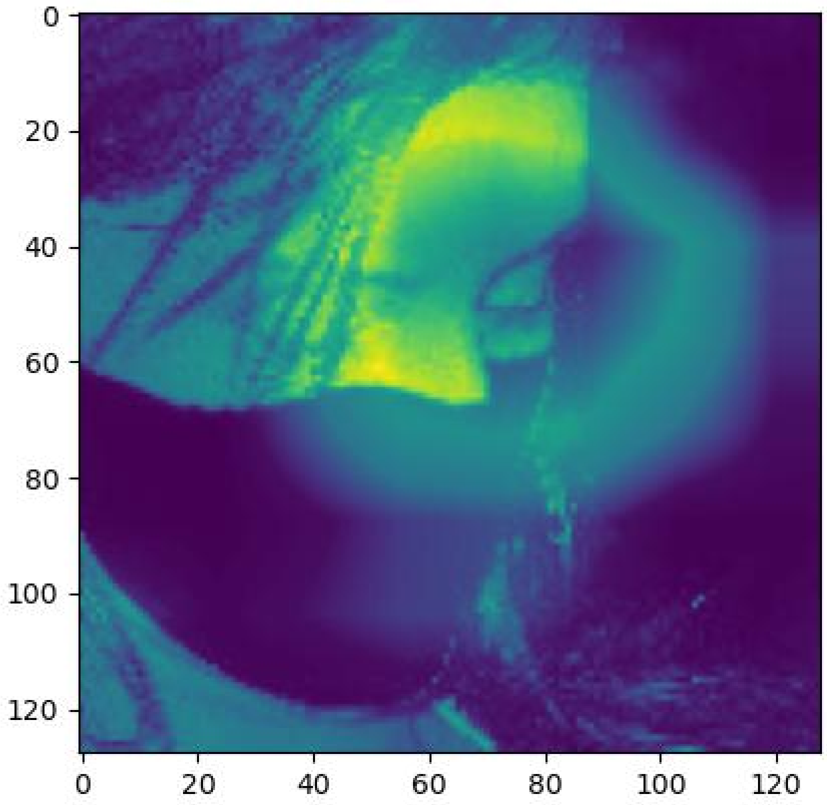


With CBAM

Grad-CAM output analyzation

By Leng Tianang





Some faces turn too much, making the network fail to focus on eyes

By Wang Rongying

MTCNN(Face alignment)

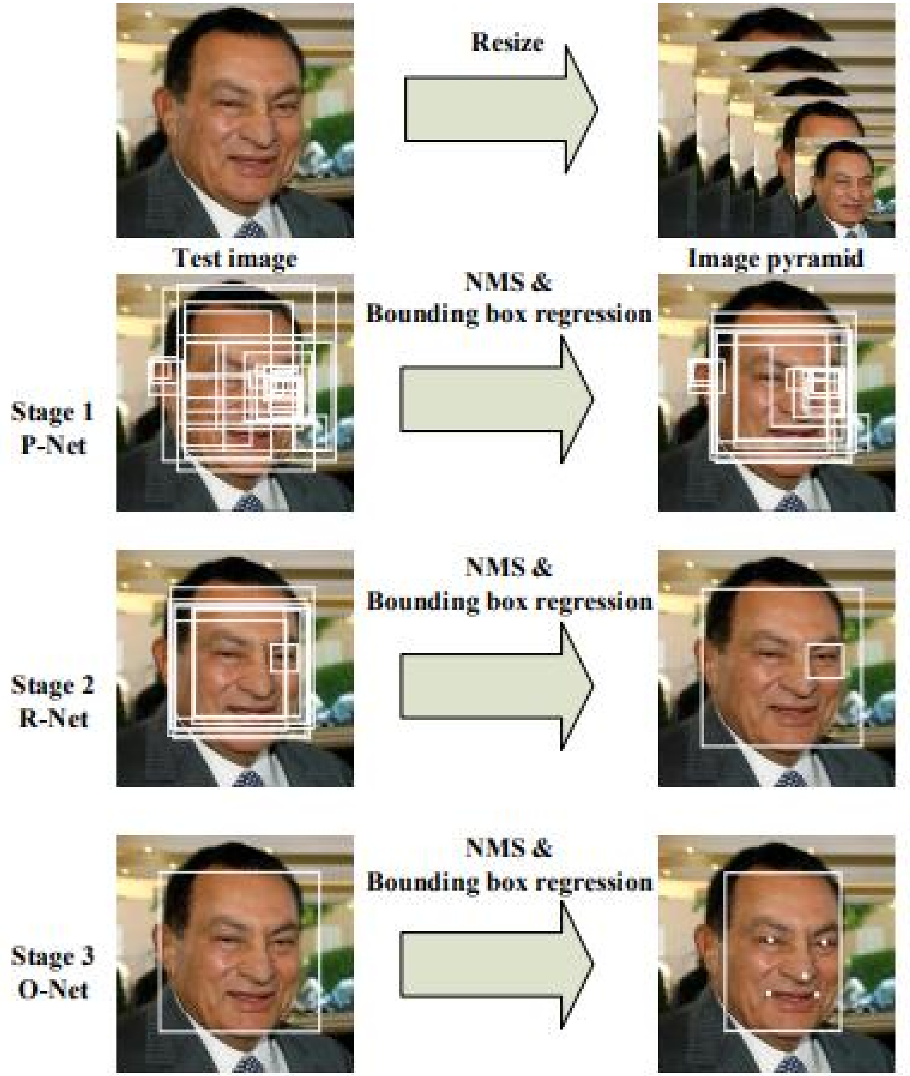
- Solve the problem of Face Detection and Facial Landmark Detection

- network structure : -- three-stage network

  Resize the original image to different sizes --> detect faces of different sizes

  1. P-Net: Obtain the candidate facial windows & NMS
  2. R-Net: Rejects a large number of false candidates & Calibration &NMS
  3. O-Net: Refined output & output five key points

$$L_i^{\text{landmark}} = \left\| \hat{y}_i^{\text{landmark}} - y_i^{\text{landmark}} \right\|_2^2$$



K. Zhang, Z. Zhang, Z. Li and Y. Qiao, "Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks," in IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499-1503, Oct. 2016, doi: 10.1109/LSP.2016.2603342.

By Wang Rongying

Cleaning dataset

- Problem:

Some side face image may cause problems to attention mechanism, make it concentrate on wrong parts.

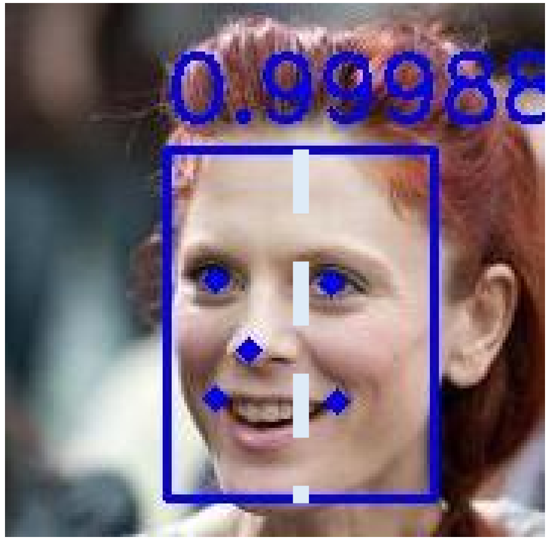
- Purpose:

Get rid of the side face data in the dataset to avoid misleading the network.

- Method:

MTCNN——obtain the positions of five key points

Detect the deviation rate of the center point of two eyes in the face recognition windows.

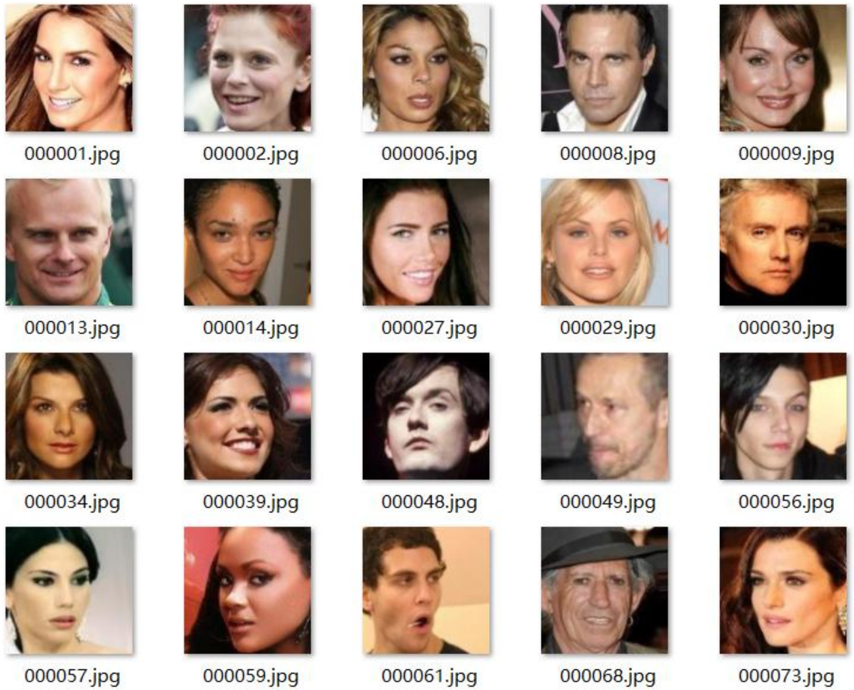


K. Zhang, Z. Zhang, Z. Li and Y. Qiao, "Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks," in IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499-1503, Oct. 2016, doi: 10.1109/LSP.2016.2603342.

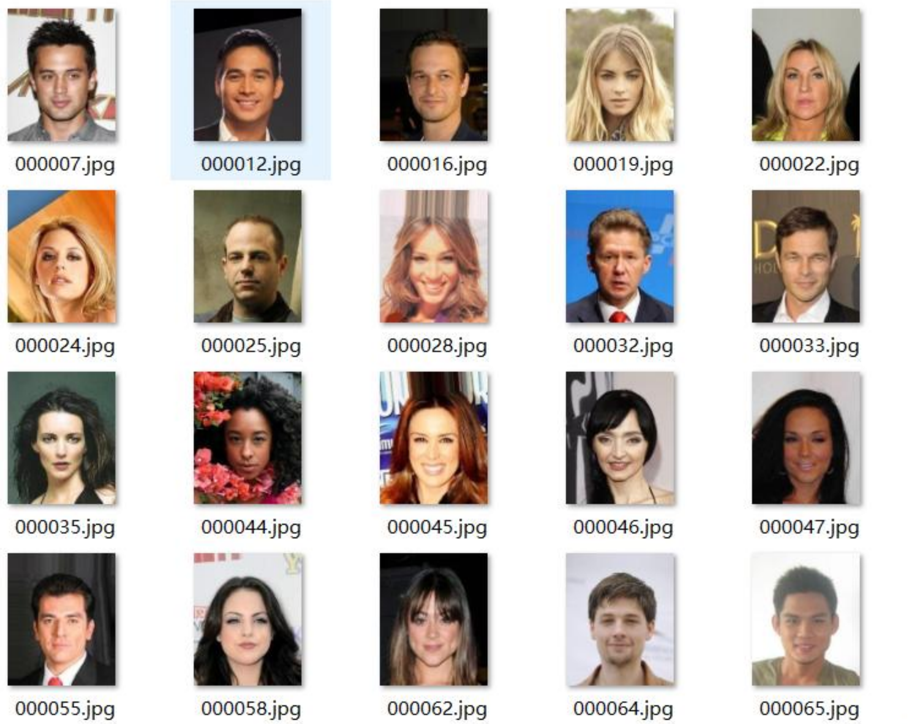
By Wang Rongying

Cleaning dataset

- Result:



Side face data example



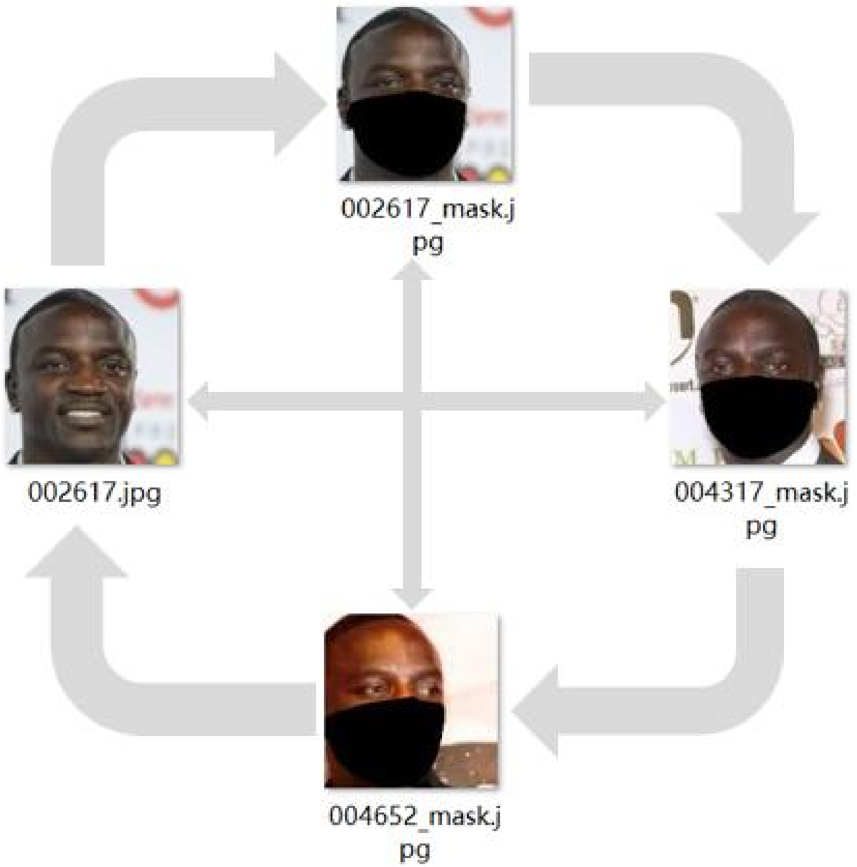
Frontal view data example

K. Zhang, Z. Zhang, Z. Li and Y. Qiao, "Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks," in IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499-1503, Oct. 2016, doi: 10.1109/LSP.2016.2603342.

By LIU YUFEI

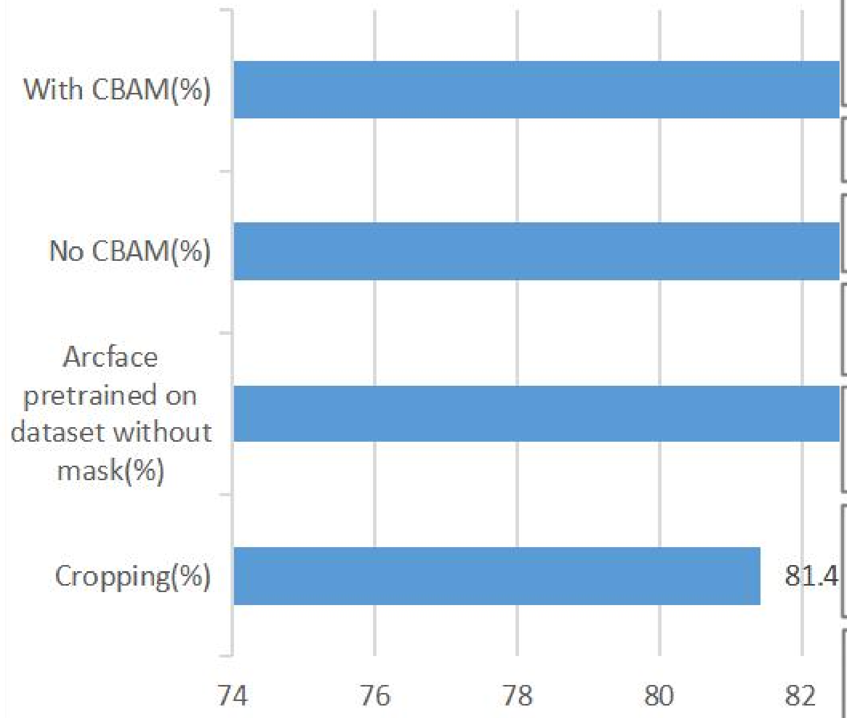
Cleaning dataset

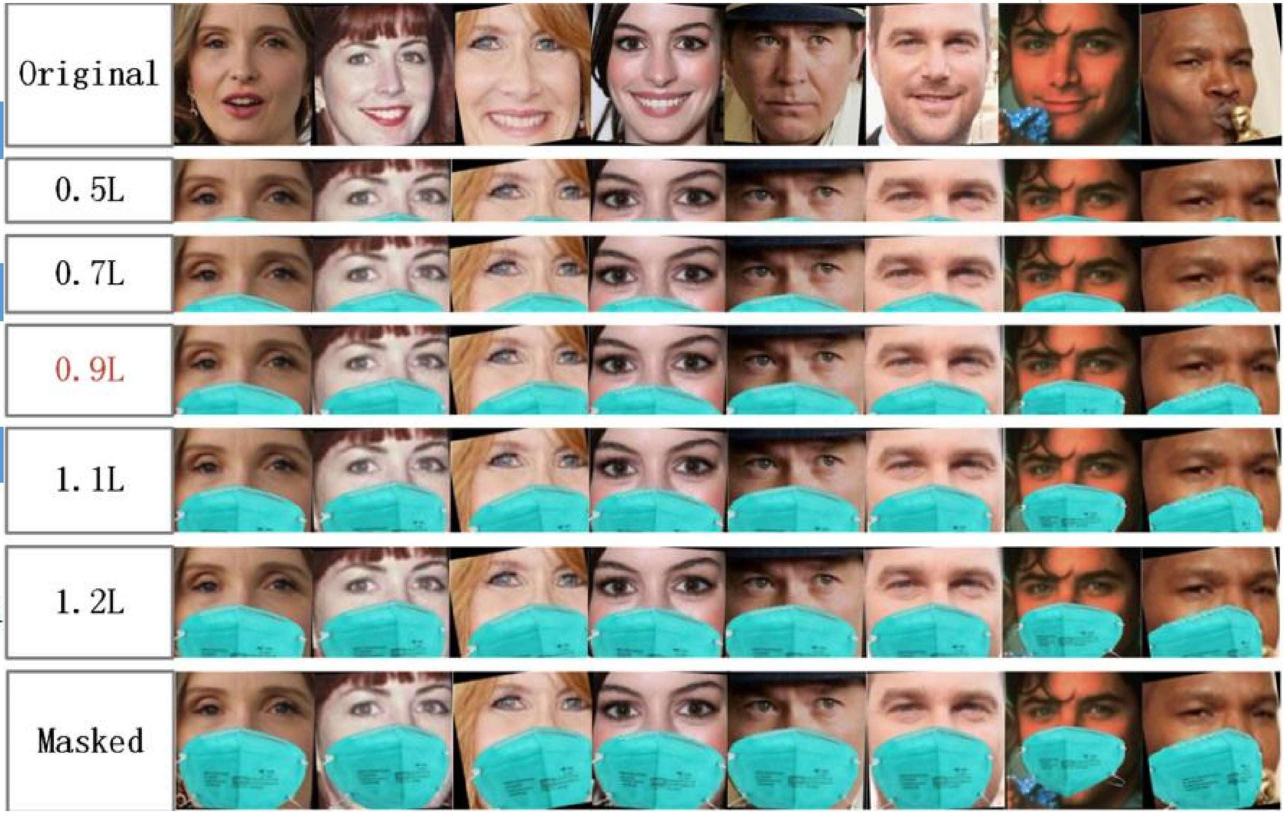
```
10000/000801.jpg 10000/000801_mask.jpg 1
10000/000801.jpg 10000/001530.jpg 1
10000/000801.jpg 10000/001530_mask.jpg 1
10000/000801.jpg 10000/008900.jpg 1
10000/000801.jpg 10000/008900_mask.jpg 1
10000/000801_mask.jpg 10000/001530.jpg 1
7995/000544.jpg 8677/006700.jpg 0
7995/000544_mask.jpg 8677/003546_mask.jpg 0
8002/003229_mask.jpg 8674/008151_mask.jpg 0
8002/003264.jpg 8674/008151.jpg 0
8014/002131.jpg 8666/000586.jpg 0
8014/002131_mask.jpg 8665/003704_mask.jpg 0
8014/002187.jpg 8665/003704.jpg 0
```



Here is how we organize our test dataset

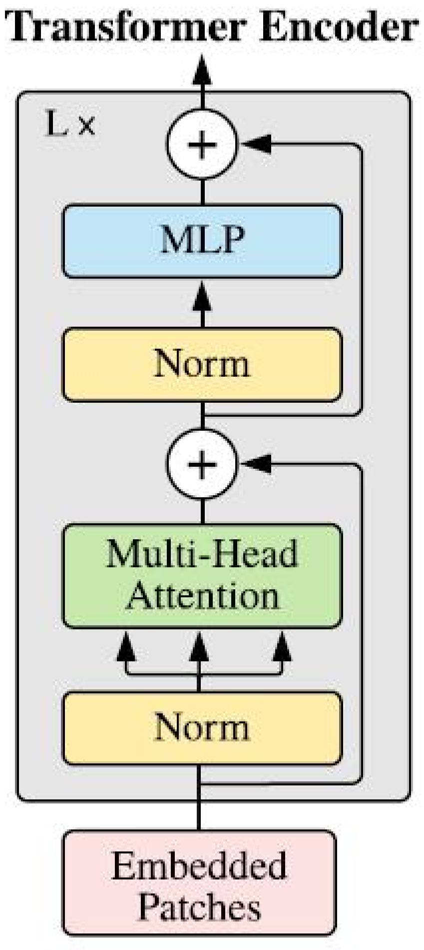
Attention Machanism ——— CBAM





Kuang Haohong

Vision Transformer

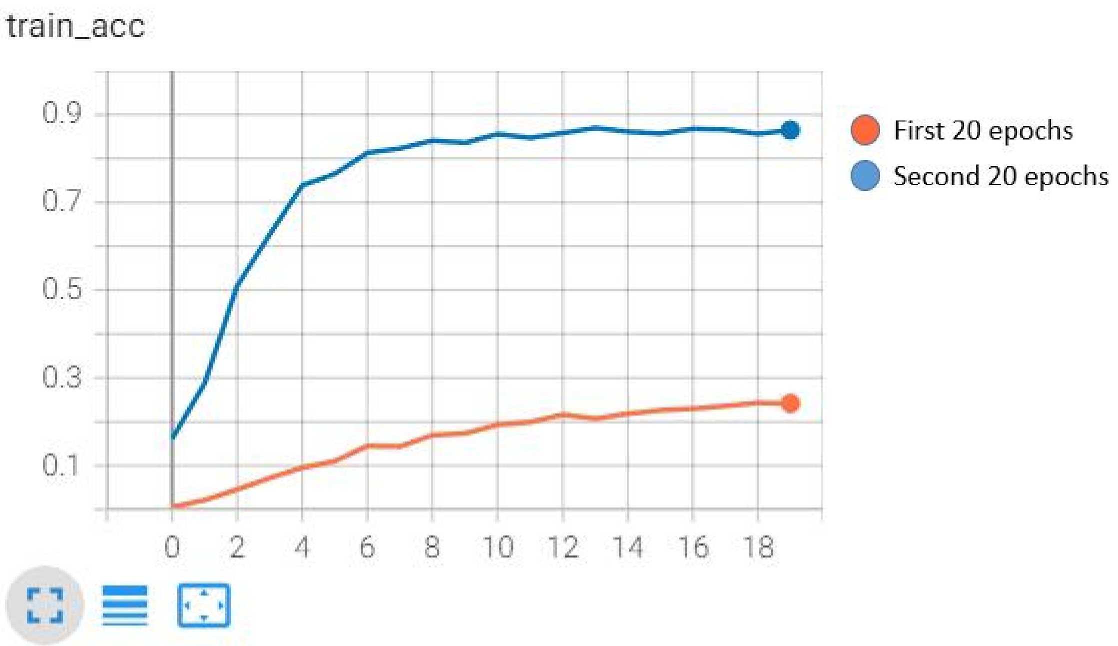


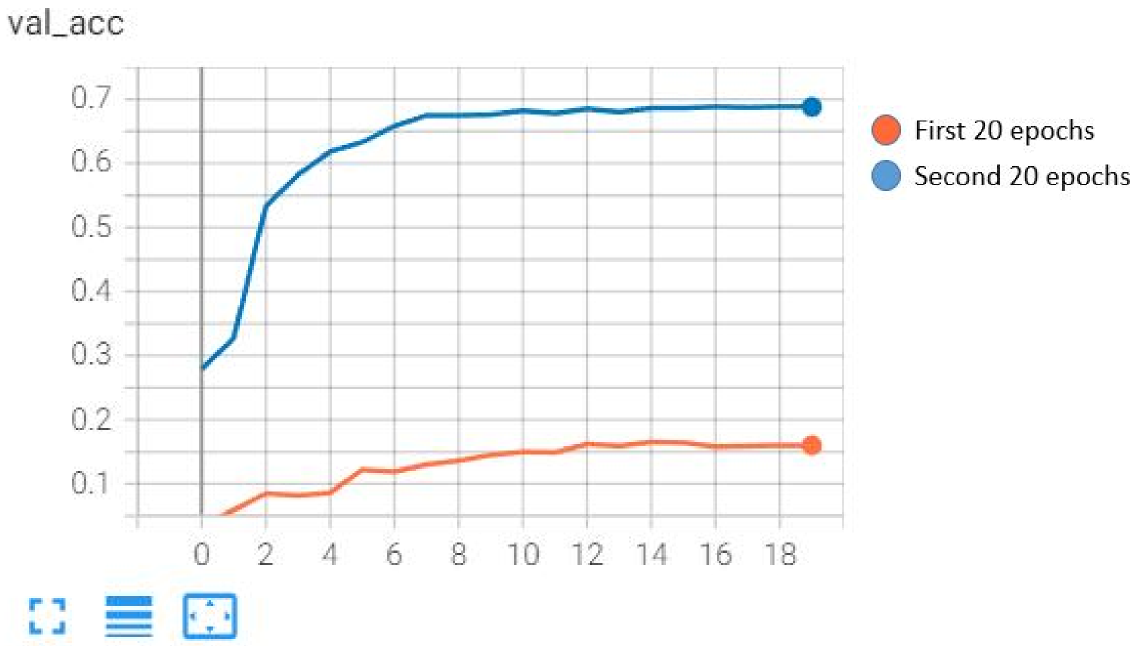
Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16x16 words: Transformers for image recognition at scale[J]. arXiv preprint arXiv:2010.11929, 2020.

Result

Kuang Haohong

The dataset: 500 categories of people, each with 10 masked-pictures





Reflection

Kuang Haohong

- The dataset is small, resulting in less than perfect performance on the vision transformer.
- Vision Transformer was inferior to CNN in feature extraction in a small number of samples.
- Hybrid Vision Transformer.

By Wang Rongying

Our harvest

- Did hands-on practice, promote our understanding of knowledge
- Learned how to do research
- Learned about the state-of-art methods in this field

By Wang Rongying

Evaluation & Future work

- Added attention to the network -> Improved accuracy
- Data Augmentation & network design

- Continue trying Vision Transformer
  - More powerful
  - Few studies have tried
- Design new loss function

We would appreciate it if you could continue to follow up our project ⭐

By Wang Rongying

Evaluation & Future work

- Added attention to the network -> Improved accuracy
- Data Augmentation & network design

- Continue trying Vision Transformer
  - More powerful
  - Few studies have tried
- Design new loss function

We would appreciate it if you could continue to follow up our project ☆

By Wang Rongying

Reference

- *Chenyu Li, Shiming Ge, Daichi Zhang, and Jia Li. Look through masks: Towards masked face recognition with deocclusion distillation. ACM Multimedia, 2020..*

- *Li, Yande, et al. "Cropping and attention based approach for masked face recognition." Applied Intelligence 51.5 (2021): 3012-3025.*

- *Chang, Wei-Yi, Ming-Ying Tsai, and Shih-Chieh Lo. "ResSaNet: A Hybrid Backbone of Residual Block and Self-Attention Module for Masked Face Recognition.“*
- *Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021.*

- *Deng, Jiankang, et al. "Arcface: Additive angular margin loss for deep face recognition." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.*

- *Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7132-7141.*

- Woo S, Park J, Lee J Y, et al. Cbam: Convolutional block attention module[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 3-19.

- *Selvaraju, Ramprasaath R., et al. "Grad-cam: Visual explanations from deep networks via gradient-based localization." Proceedings of the IEEE international conference on computer vision. 2017.*

- *K. Zhang, Z. Zhang, Z. Li and Y. Qiao, "Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks," in IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499-1503, Oct. 2016, doi: 10.1109/LSP.2016.2603342.*

- *Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16x16 words: Transformers for image recognition at scale[J]. arXiv preprint arXiv:2010.11929, 2020.*

Thank you for listening