Supplementary Materials

A Details of Experiment Setup

A.1 Self-supervised pre-training

Hyper-parameters. All the self-supervised learning methods applied ResNet 50 as the backbone, followed by two MLP layers that project the embedding feature into a 128-dimension space. The batch size was set as 256. We optimized the network by Adam with the weight decay as 1e-4 and learning rate initialized as 3e-4. We applied cosine annealing learning rate scheduler from 10 epochs onwards. Empirically, MoCo-v2 and BYOL were optimized by SGD instead.

Data augmentation. Details of the data augmentation strategies in our proposed contrastive learning are shown in Algorithm 1. Below are the explanations of the augmentation operations we applied.

- **TimeAug**: Randomly sample along time domain with t_1 time interval, which follows a downscale distribution over $[0, T_1]$. In our case, $T_1 = 1$ second.
- **FaceSwap**: Randomly select an image from the training set and set it as the sample that provides identity information S_{id}. Apply FaceSwap to swap facial expression of x. The probability of performing FaceSwap p_s follows a Bernoulli distribution with probability as 0.5.
- **Mask**: Randomly apply mask on the area of eyes and mouth with the normalised value. Probability p_m follows a Bernoulli distribution (0.8).
- **Resize**: Resize the image to a size of 128×128.
- **Crop**: Randomly crop a region from the image with a size of 112×112.
- **Flip**: Horizontal flip, with a probability p_f of 0.5
- **Jitter**: Colour jittering (0.4 brightness, 0.4 contrast, 0.4 saturation and 0.2 hue). Probability p_j is set to 0.8
- **Blur**: Gaussian blur, with a probability p_b under 0.5 Bernoulli distribution.
- **Gray**: Grayscale, with a probability p_g of 0.5 Bernoulli distribution.

Algorithm 1 Data augmentation for self-supervised pre-training.

Input: Images $X = \{x_1, x_2, ...x_N\}$ from video clips
for $x \in X$ **do**
  if x is positive **then**
    x'=TimeAug(x, t_i) where $t_1 \in T_1$
    x'=FaceSwap(x') if p_s
  else if x is anchor **then**
    x'=x
  end if
  x'=Mask(x') if p_m
  x'=Crop(Resize(x'))
  x'=Flip(x') if p_f
  x'=Jitter(x') if p_j
  x'=Blur(x') if p_b
  x'=Gray(x') if p_g
end for
Output: Augmented images $X' = \{x'_1, x'_2, ...x'_N\}$

A.2 Downstream task

A.2.1 Facial expression recognition

Hyper-parameters. We provide results of both freezing (freezing the pre-trained model layers) and fine-tuning (tuning all layers). All downstream tasks of FER (Emotion Classification and Valence & Arousal Recognition) were trained with a batch size of 64. The network was trained with an Adam optimizer with the weight decay as 5e-4 over 20 epochs for AffectNet dataset and 300 epochs for FER2013 dataset. Initially, the learning rate was set to 1e-4 and decay with cosine annealing learning rate scheduler.

It should be noted we evaluated the pretrained model of CycleFace [1] with a different set of hyper-parameters, empirically, on the emotion classification task of FER2013. We optimized the cross-entropy loss using SGD with Nesterov momentum, using a batch size of 64, a weight decay of 1e-4 and a momentum of 0.9. The learning rate was decayed using Reduce learning rate on Plateau scheduler with the initial learning rate of 1e-2.

Data augmentation. The applied data augmentation of downstream task training, is illustrated in Algorithm 2. It should be noted that when fine-tuning with CycleFace [1], we resized the images to (80, 80) and then cropped to (64, 64) because the network only accepts this size of image input.

Algorithm 2 Data augmentation for downstream task.

Input: Images $X = \{x_1, x_2, ...x_N\}$ from video clips
for $x \in X$ **do**
  x'=Mask(x) if p_m
  x'=Crop(Resize(x'))
  x'=Flip(x') if p_f
  x'=Jitter(x') if p_j
  x'=Blur(x') if p_b
  x'=Gray(x') if p_g
end for
Output: Augmented images $X' = \{x'_1, x'_2, ...x'_N\}$

A.2.2 Face recognition

We also tested our results on face recognition with KNN to further validate that our method is more robust against other facial attributes, such as face identity. The distance between features is calculated using $L2$ norm, with the same setting as Cycleface [1].

B Additional Results

B.1 Computational cost

Methods	batchsize	time/150-epochs	memory/GPU
SimCLR [2]	256	8h	1.6GB
Ours (All Strategies)	256	17.5h	2.2GB

Table S1: Comparison of time and memory cost between SimCLR and Ours (with Nvidia RTX 3090).

We present the computational cost of our proposed method in Table S1. This was measured under the same experimental settings (e.g., hardware and batchsize). It can be observed that under the same epoch number, it would take more time and more GPU memory for our

proposed method. Moreover, as shown in Table 3 in the main paper, SimCLR tends to learn short-cuts when the epoch number is still small yet the Top1-instance classification is high. With our proposed series of strategies, although the computation cost is larger, our proposed method can effectively avoid shortcuts such as face identity.

On the other hand, we argue that the proposed FaceSwap is an effective strategy to avoid identity-related shortcuts, by generating intermediate fake faces during the training stages. This simple operation may introduce some artifacts (as shown in Figure 3 of the main paper) in the face-swapped images, compared to those state-of-the-art Deepfake algorithms [3, 4]. However, it is much more computationally efficient for online data augmentation.

B.2 Batchsize sensitivity

Methods	batchsize	FER-Finetune	
		F1↑	Acc↑
Ours (TimeAug+HardNeg+MaskFN)	64	56.4%	56.5%
Ours (TimeAug+HardNeg+MaskFN)	256	58.9%	58.9%

Table S2: Pre-trained model with different batchsize.

We present the results of applying different batchsize during pretraining. As shown in table S2, reducing the batchsize would impair the performance. We think this issue could come from two perspectives:

- Contrastive learning itself is sensitive to batchsize since the core InfoNCE loss applied in contrastive learning has been proven to benefit from large batch sizes [2].
- Our method for False Negative Cancellation was designed based on the categorical expression assumption, as discussed in the Section 4.6 of the main paper. That is, when the batch size is much larger than the downstream category number, the facial images with similar mouth-eye descriptors have higher chances of being the same category. Therefore, a larger batchsize would increase the probability of correctly picking up false-negative samples.

References

[1] Jia-Ren Chang, Yong-Sheng Chen, and Wei-Chen Chiu. Learning facial representations from the cycle-consistency of face. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9680–9689, 2021.

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[3] Ivan Perov, Daiheng Gao, Nikolay Chervoniy, Kunlin Liu, Sugasa Marangonda, Chris Umé, Mr Dpfks, Carl Shift Facenheim, Luis RP, Jian Jiang, et al. Deepfacelab: Integrated, flexible and extensible face-swapping framework. *arXiv preprint arXiv:2005.05535*, 2020.

[4] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1–11, 2019.