

## APPENDIX

In the appendix, we provided the supplemented materials, including data preprocessing in Sec. A. In Sec. B, we described model pre-training for contrastive learning, model training for single attribute and MaSS. We included the ablation studies in Sec. C and how to generate compared results in Table 5 and Table 6 in Sec. D. Our source code and models will be publicly available to help better understand the settings of training and evaluation.

### A DATA PREPROCESSING

As briefly discussed in the main manuscript, MaSS takes a feature vector in and then generates a feature vector instead of operating on the raw data. Therefore, for each dataset, we convert the data into feature vectors via state-of-the-art neural networks and then normalize the vector by its L2-norm.

**Adience.** We first resize the image into  $160 \times 160$  and normalize the image by the mean and the standard deviation used in the FaceNet (Schroff et al., 2015). Then, we feed the normalized image into FaceNet to get a 512-d feature vector.

**AudioMNIST.** The majority of the information in audio signal resides at the beginning, and the average length of a waveform is 30,844 samples and the upper quartile is 34,380. Therefore, we either truncate and pad (zeros) the waveform to the length of 30,000 at the end such that the data loader can form them as a batch to speed up the training. Then, we feed the truncated/padded waveform into Hubert-L (Hsu et al., 2021) to get a 1024-d feature vector after performing average pooling on the output of Hubert-L along the time dimension.

**VISPR and PA-HMDB.** R3D-18 is trained with the clip size of  $16 \times 112 \times 112$  and generates a 512-d feature vector; therefore, we resize the spatial dimension of a video into  $112 \times 112$  and then sample 16 frames (every other frame) out of a video to form a clip. For VISPR, since it is an image dataset, we generate a 16-frame clip by duplicating the same image and then pass it to R3D-18 to extract features. On the other hand, for the action attribute in PA-HMDB, we convert each video into frame-level feature vectors by R3D-18. More specifically, for each timestamp, we take its neighboring frame (every other frames) to form a 16-frame clip and then feed it to R3D-18. E.g., for a video with 100 frames, we will get 100 512-d feature vectors. For the other attributes in PA-HMDB, since those labels are image-level instead of video-level, we simply assign the labels to the frame-level features extracted above.

### B MODEL TRAINING

#### B.1 ATTRIBUTE-AGNOSTIC MODEL PRE-TRAINING

For all datasets, we follow similar practices to train attribute-agnostic models via SimCLR (Chen et al., 2020). First, we generate two views of data by different data augmentation in the raw data domain, and then the two views of data are passed through the fixed feature extractor to get its feature representation. After that, we train a multi-layer perceptron (MLP) as an encoder to learn a generic feature representation over the features extracted by the fixed feature extractor. The MLP is composed of two fully-connected layers with the same dimension as the input feature dimension, and 1-D batch normalization layer and ReLU are added between fully-connected layers. The trained MLP is served as attribute-agnostic model in MaSS. Note that we also adopt the MLP-projector in the contrastive learning to achieve better performance. We train the model for 100 epochs with temperature 0.07 via the stochastic gradient decent (SGD) optimizer. The weight decay is set to 0.0001 and the learning rate starts from 0.05 and then it is annealed with cosine schedule. In the follow paragraphs, we describe how to generate different views for each dataset.

**Adience.** To generate two views, we first resize images to  $160 \times 160$  and then randomly flip the image horizontally; after that, we randomly perform color jitter via torchvision package with 80% probability and then convert the image into gray scale with 20% probability.

Table 7: Model training settings for each dataset in Table 2, 3 and 4.

Dataset	#GPUs	Batch Size per GPU	Learning Rate	Loss Weights
Adience	1	64	0.00125	$w_{\text{rec}} = 1, w^{s_1} = 5, h^{s_1} = 1$ $w^{r^*} = 40, w^{r_1} = 1, w^{r_2} = 1$
AudioMNIST	4	128	0.01	$w_{\text{rec}} = 10, w^{s_1} = 0.001, h^{s_1} = 0.1$ $w^{r^*} = 15, w^{r_1} = 0.01, w^{r_2} = 0.001, w^{r_3} = 0.001$
PA-HMDB	4	128	0.01	$w_{\text{rec}} = 1, w^{s_1} = 1, h^{s_1} = 0.1$ $w^{r^*} = 1000, w^{r_1} = 50$

Table 8: Results on Adience with different suppressed attribute, the experiments are completed under the loss setting of  $L_{\text{rec}} + L^{s_1} + L^{r^*}$ , where  $s_1$  is the suppressed attribute.

Suppressed Attribute		Top-1 Accuracy (%)		
		DataID	Age	Gender
MaSS	DataID	0.6	78.5	95.7
	Age	81.2	13.7	94.5
	Gender	77.2	75.3	8.3

**AudioMNIST.** In this work, we apply two different augmentations (Ma, 2019): random crop on the entire audio with a coverage of 0.4 and mask with a coverage of 0.5 to each view respectively. The crop augmentation removes the selected part from the audio, whereas mask substitutes it with zeros.

We limited the data augmentations used in our methods to crop and mask because other augmentation like pitch, loudness, speed, etc. would affect the structure of audio signal and potentially won't be able to retain attributes like gender, accent, and age.

**PA-HMDB.** We generate two views of data by following the practice in CVRL (Qian et al., 2021), i.e., for a positive pair, two views are extracted from different time instance of a video and the temporal-consistent data augmentation is performed on each view. The data augmentation is composed of resizing the spatial dimension into  $112 \times 112$ , gaussian blurring, randomly converting color image into gray image.

## B.2 ATTRIBUTE-SPECIFIC MODEL PRE-TRAINING

For all attributes in all datasets, we train the attribute-specific model by using the cross-entropy loss against the given label to compute the gradient for all parameters through back-propagation. The model contains three fully-connected layers and with the dimension: *input dim-512-256-number of classes*, and the 1-D batch normalization layer and ReLU are added between layers. We use a batch size of 256 with the AdamW optimizer (Loshchilov & Hutter, 2019) to train the model for 100 epochs. The weight-decay is fixed to 0.05 and the initial learning rate is set to 0.01 and then the learning rate is annealed with the cosine scheduler.

## B.3 MASS TRAINING

The training on different datasets follows similar settings but with different loss weights. When training the data modifier  $G$  in MaSS, all models in the suppression and preservation branches are fixed without any update. We train all models with 100 epochs with the AdamW optimizer (Loshchilov & Hutter, 2019). The weight-decay is 0.05 and we adopt cosine learning scheduler to anneal the learning rate. For the loss type, in most of cases, we use cosine similarity measurement for the to-be-suppressed attribute and KL divergence for the attribute-specific preservation. Table 7 described other training details for different datasets.

Table 9: Results on Adience with different weights on  $L^{r^*}$ , the experiments are completed under the loss setting of  $L_{rec} + L^{s_1} + L^{r^*}$ . We used  $w^{r^*} = 40$  in our main results.

	$w^{r^*}$	Top-1 Accuracy (%)		
		DataID	Age	Gender
MaSS	10	0.0	69.4	73.5
	20	0.0	75.0	94.8
	40	0.6	78.5	95.7
	80	3.2	82.1	96.4
	160	13.5	83.5	96.4

Table 10: Results on Adience with different similarity measurements in  $L^{s_1}$ , the experiments are completed under the loss setting of  $L_{rec} + L^{s_1} + L^{r^*}$ .  $s_1$  is DataID.

	Similarity	Top-1 Accuracy (%)		
		DataID	Age	Gender
MaSS	Cosine	0.6	78.5	95.7
	KL divergence	0.2	76.2	95.8
	CE	0.1	76.7	94.9

Table 11: Results on Adience under multiple suppressed attributes. The DataID and gender are selected to be suppressed. MaSS is configured with  $L^{r^*}$  and  $L^{r_1}$ .

	Top-1 Accuracy (%)		
	DataID ( $s_1$ )	Gender ( $s_2$ )	Age ( $r_1$ )
Original	90.8	97.4	89.1
MaSS	0.9	5.0	83.8

## C ABLATION STUDIES

In ablation studies, we use the Adience dataset for all experiments and we discuss MaSS in three perspectives, including suppressing different attributes, effects of loss weights, effects of similarity measurement.

**Suppression Target.** In the main manuscript, we always suppress DataID in all experiments; however, MaSS is configurable to suppress any attribute while still preserving others. Table 8 shows the results by suppressing different attributes. Only the performance of the selected attribute is degraded while other attributes are still good. Note that those results do not include any attribute-specific models in the preservation branch. The result shows that MaSS is flexible to configure to suppress any attribute and preserve others.

**Loss Weights.** Intuitively, the loss weight controls which loss term should be focused on more during the optimization. In this ablation study, we vary the weights for the attribute-agnostic model and the results are shown in Table 9. Since  $L^{r^*}$  controls how generic the feature representation is, the higher weights preserve more generic features; therefore, the transformed dataset could perform better for all attributes. However, when  $L^{r^*}$  is 160, the accuracy of DataID is also increased because the strength of suppression is not strong enough since the weight of  $L^{r^*}$  is too high.

**Different Similarity Measurement for Suppression.** We proposed three different measurements for the similarity in the main manuscript. Those measurements provided similar functionalities conceptually but they might work different empirically. Table 10 shows the results with different measurements, and all results are close to each other. Therefore, for suppression, we use cosine for all experiments.

**Multiple Suppressed Attributes.** Table 11 shows the results when we configured MaSS to suppress multiple attributes, DataID and gender. The accuracy of both DataID and gender attributes are

Table 12: Ablation studies of losses in the suppression branch on Adience. DataID is selected as the suppression target.

	Loss Configuration			Top-1 Accuracy (%)				
	$L^{s_1}$	$L_{rec}$	Entropy Loss	L2 Loss	Entropy	DataID ( $s_1$ )	Gender	Age
MaSS	✓	-	-	0.003	5.4	0.0	28.6	68.9
	✓	✓	-	0.002	5.4	0.0	27.4	68.1
	✓	✓	✓	0.002	6.6	0.0	33.8	73.5

Table 13: Results on Adience under different configurations. DataID is selected as the suppression target.

	Loss Configuration					Top-1 Accuracy (%)		
	$L_{rec}$	$L^{s_1}$	$L^{r_s}$	$L^{r_1}$	$L^{r_2}$	DataID ( $s_1$ )	Age ( $r_1$ )	Gender ( $r_2$ )
Original	-	-	-	-	-	90.8	89.1	97.4
MaSS	✓	✓	✓	✓	✓	0.6	86.9	96.7
	✓	✓	-	✓	✓	0.0	84.5	96.3

Table 14: Results on AudioMNIST under different configurations. The checkmark (✓) denotes that the particular loss term is used in the optimization. SpeakerID is selected as the suppression target.

	Loss Configuration						Top-1 Accuracy (%)			
	$L_{rec}$	$L^{s_1}$	$L^{r_s}$	$L^{r_1}$	$L^{r_2}$	$L^{r_3}$	SpeakerID ( $s_1$ )	Digit ( $r_1$ )	Accent ( $r_2$ )	Gender ( $r_3$ )
Original	-	-	-	-	-	-	95.6	99.8	99.3	96.5
MaSS	✓	✓	✓	✓	✓	✓	1.7	99.6	95.7	98.4
	✓	✓	-	✓	✓	✓	1.6	99.7	95.0	97.7

Table 15: Results on VISPR and PA-HMDB under different configurations. The checkmark (✓) denotes that the particular loss term is used in the optimization. Metrics for the action attribute is Top-1 Accuracy (%) while cMAP (%) is used for the other 5 non-action attributes. MaSS is configured to suppress the non-action attributes.

	Loss Configuration				VISPR		PA-HMDB	
	$L_{rec}$	$L^{s_1}$	$L^{r_s}$	$L^{r_1}$	Non-action Attrs. ( $s_1$ )		Action ( $r_1$ )	Non-action Attrs. ( $s_1$ )
Original	-	-	-	-	81.8		58.7	79.7
MaSS	✓	✓	✓	✓	38.6		58.0	63.4
	✓	✓	-	✓	38.3		52.2	63.6

suppressed successfully but the age attribute can still be recognized even without age labels. With age label, MaSS can further improve its accuracy.

**Losses in Suppression Branch.** Here, we further explore the effects of L2 reconstruction loss and prediction entropy loss applied in the suppression branch.

We add the L2 reconstruction loss to minimize changes in the data while suppressing the targeted attributes; thus, we are able to keep the data as truthful as possible. The prediction entropy loss is added to increase the entropy of the predicted probability such that the prediction of the modified data becomes closer to a random guess; thus, there is less information in the prediction.

Table 12 shows the effects of these two loss terms when only suppression is considered, i.e., no preservation branch. With L2 reconstruction loss, the overall performance of attribute recognition does not affected; however, it reduced the overall L2 loss. After adding prediction entropy loss, the accuracy of DataID is still 0% but its entropy is increased, which makes the prediction closer to a random guess, as desired.

Table 16: Comparison to SPAct (Dave et al., 2022), Metrics for the action attribute is Top-1 Accuracy (%) while cMAP (%) is used for the other 5 non-action attributes. Non-action attributes are suppressed in the experiment, and MaSS is configured without  $L^r$ .

	VISPR		PA-HMDB
	Non-action Attrs. ( $s_1$ )	Action ( $r_1$ )	Non-action Attrs. ( $s_1$ )
Original	81.8	58.7	79.7
MaSS	36.3 (↓55.6%)	52.1	63.2 (↓20.7%)
Original (SPAct)	64.4	-	70.1
SPAct	27.4 (↓57%)	-	58.9 (↓16%)

**Effects of  $L^r_*$  with All Labels.** In Table 2, 3 and 4, we have shown that the benefits to have attribute-agnostic loss ( $L^r_*$ ) compared to suppression only. Here, we discussed the contribution of  $L^r_*$  when all to-be-preserved attributes are available, and the results are shown in Table 13, 14 and 15. Without  $L^r_*$ , it achieved competitive performance on to-be-preserved attributes to the one with  $L^r_*$  on Adience and AudioMNIST for all attributes. Moreover, for PA-HMDB, without  $L^r_*$ , the accuracy of action is degraded 5.8%, we conjecture that even though the generic feature retained by enforcing  $L^r_*$  help the data utilities of the modified data.

## D COMPARED RESULTS

**Adience.** We compared many approaches in Table 5 and here we describe the details for how to generate those results. First, for Gaussian noise, we added zero-mean with different standard deviations ( $\sigma$ ) into the original feature vectors to manipulate data. For Gaussian blur, downsample and obfuscation are all performed in the raw data domain, and then the modified data are passed through FaceNet to get the feature representation. For Gaussian blur, we apply zero-mean with various standard deviations ( $\sigma$ ) with different kernel sizes ( $k$ ) to blur the image. For downsample, we downsample the data with different ratios and then upsample it back to original size. Lastly, for obfuscation, we use MTCNN to detect the location of the face and then remove the face region with different ratios.

For CIAGAN (Maximov et al., 2020), we first followed CIAGAN’s method to pre-extract the masked face and the facial landmark information for the Adience dataset by using the Dlib-ml library (King, 2009). And then, the CIAGAN model takes in the Adience images, their landmarks, masked faces and the desired target. For DeepPrivacy (Hukkelås et al., 2019) and Fawkes (Shan et al., 2020), we adjusted the released codes and ran over the Adience dataset to generate new images.

After we obtain the transformed Adience images, we use the same procedure as ours for evaluation: using FaceNet (Schroff et al., 2015) to extract the feature vector of an image.

**AudioMNIST.** We compared two methods in AudioMNIST, including adding white noise and masking out a portion of waveform based on the nlpaug library (Ma, 2019). We use the default parameter for white noise and set the masking ratio to 50% of the waveform.

**SPAct.** We tried our best to compare with SPAct (Dave et al., 2022) under the same experimental condition; however, our absolute performance over the raw data is significantly better than their paper; therefore, we only compare the relative gains to them. Moreover, under this setting, they do not show the accuracy of the action attribute. Table 16 shows the comparison to SPAct and baselines. When comparing to its own baseline, our method is competitive in suppressing non-action attributes while keeping good accuracy in the action attribute.