# Supplementary Document of Unimodal Face Classification with Multimodal Training

Wenbin Teng[1] and Chongyang Bai[2]
[1] Boston University
[2] Dartmouth College

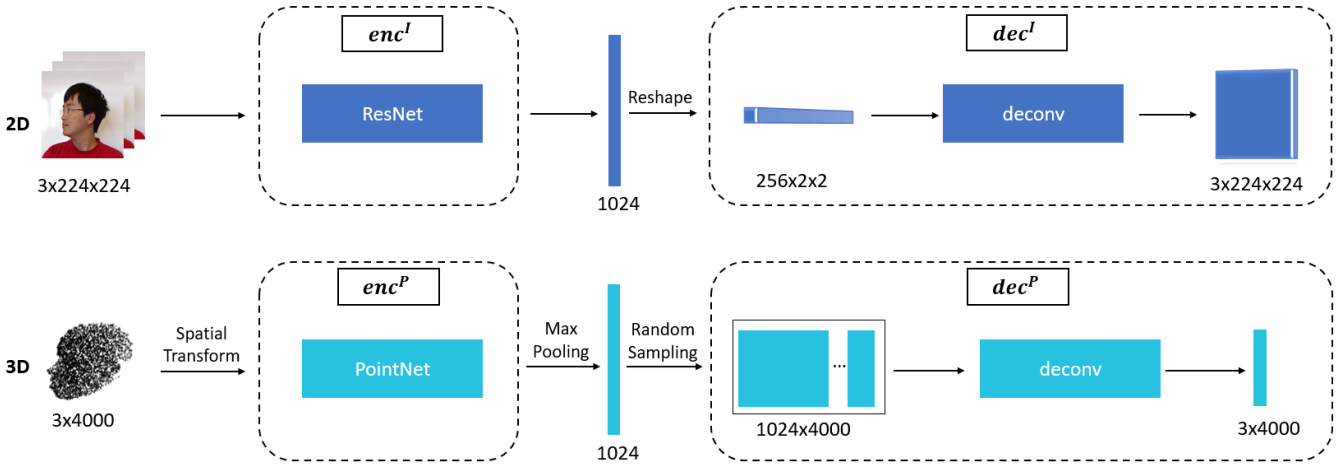


Fig. 1: An overview of 2D (top row) and 3D (bottom row) autoencoder networks: 2D encoder applies ResNet [4] and 3D applies PointNet [5]. Both 2D and 3D decoder networks apply multiple deconvolution layers. The decoder networks are trained to reconstruct the missing modality in testing mode. For example, if 3D point cloud is missing during testing, 3D decoder is trained to reconstruct 3D point cloud.

Our supplementary document is organized as follows. We introduce the details of our 2D and 3D autoencoders in Section I and Section II. Section III visualizes some other evaluation results and ablation studies in addition to our main article. Section **??** are among the main successful works that inspire our idea.

## I. 2D Autoencoder

The upper left of Figure 1 represents the details of our 2D encoder: $enc^I$. We use ResNet-18 [4] as our backbone to extract face embedding. The 2D decoder $dec^I$ is shown on the upper right of Figure 1. $dec^I$ consists of several 2D transposed convolution layers. Table I shows the network configuration of our decoder network. We apply 6 deconvolution layers with different kernels to reconstruct face embedding into images with size of $224 \times 224 \times 3$; a 2D batch normalization layer is also added after each deconvolution layer; ReLU activation is added afterward. The output of $dec^I$ is finally fed into a Sigmoid function to ensure that the output of 2D autoencoder is within the range of $[0, 1]$ same as input image.

## II. 3D Autoencoder

The bottom left of Figure 1 shows our 3D encoder details: $enc^P$. We employ the PointNet architecture [5] on 3D encoder denoted as $enc^P$. Suppose the inputs are $F$-dimensional point features with $n$ points, denoted by $\mathbf{P} = \{\mathbf{P}_1, \ldots, \mathbf{P}_n\}$, $\forall i, \mathbf{P}_i \in \mathbb{R}^F$. $F = 3$ as we use $(x, y, z)$ coordinates for point features. The PointNet network regards a point cloud as a direct graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ to represent the whole point cloud structure, where $\mathcal{V}$ represents vertices of the graph, which are the points, and $\mathcal{E}$ represents the edge information obtained through $k$-nearest neighbors ($k$-NN) measured by point distances in point cloud $\mathbf{P}$.

First, the encoder learns a spatial transform to project the input point cloud to a desired space. Then, three 1D CNN

| Layer | Kernel | Stride | Output Size |
|---|---|---|---|
| Deconv-1 | $7 \times 7$ | 2 | $7 \times 7 \times 256$ |
| BN-1 | | | $7 \times 7 \times 256$ |
| Deconv-2 | $4 \times 4$ | 2 | $14 \times 14 \times 128$ |
| BN-2 | | | $14 \times 14 \times 128$ |
| Deconv-3 | $4 \times 4$ | 2 | $28 \times 28 \times 128$ |
| BN-3 | | | $28 \times 28 \times 128$ |
| Deconv-4 | $4 \times 4$ | 2 | $56 \times 56 \times 64$ |
| BN-4 | | | $56 \times 56 \times 64$ |
| Deconv-5 | $4 \times 4$ | 2 | $112 \times 112 \times 32$ |
| BN-5 | | | $112 \times 112 \times 32$ |
| Deconv-6 | $4 \times 4$ | 2 | $224 \times 224 \times 3$ |
| Sigmoid | | | $224 \times 224 \times 3$ |

TABLE I: The network structure of 2D decoder. Note that we also apply a ReLU layer after each convolution transpose layer section.

978-1-6654-3176-7/21/$31.00 ©2021 IEEE

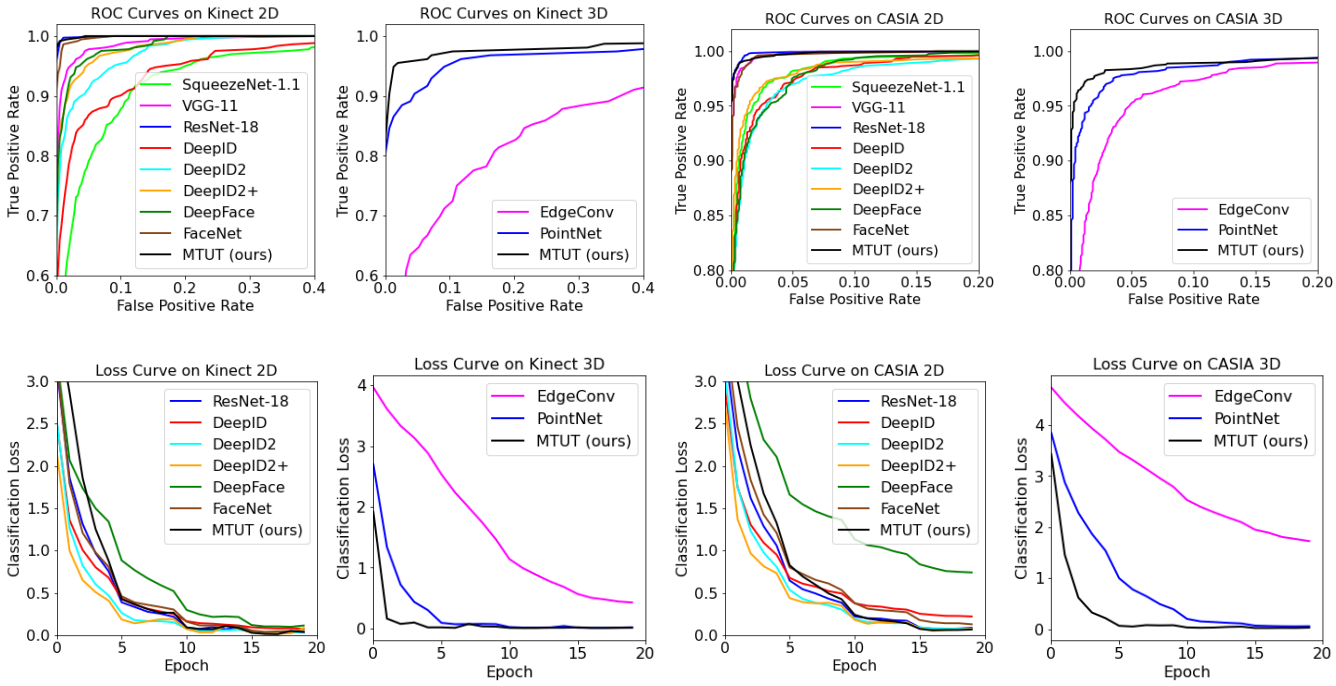


Fig. 2: ROC curves (top row) and loss curves (bottom) row comparison between proposed method and baseline models on both datasets. Please note that for better visualization loss curves of VGG-11 and SqueezeNet-1.1 are not shown.

layers are applied. Suppose $\mathbf{P}_i^l, i = \{1, 2, \ldots, n\}$ is the input of graph convolution layer $l$, the output of layer $l$ as:

$$\mathbf{P}_i'^l = h_\Theta\left(\mathbf{P}_i^l\right), l = 1, 2, ...L \tag{1}$$

where $h_\Theta : \mathbb{R}^F \rightarrow \mathbb{R}^{F'}$ is a non-linear function with a set of learnable parameters $\Theta$. $F'$ is the output dimension of graph convolution network. The spatial transform block learns a $F \times F$ transformation matrix from features of input point cloud. The transformation of original points will project them into a space facilitating the classification task. After spatial transformation and several convolution layers, PointNet architecture extracts 1024-dimension node embedding for each vertex $v \in \mathbf{P}$. Finally, max pooling is applied to the node embeddings to obtain the final latent embedding: $\mathbf{x}^P \in \mathbb{R}^{1024}$.

Our 3D decoder $dec^P$ is illustrated on the bottom right of Figure 1. The last max-pooling operation for $enc^P$ is hard to reverse as the pooling is operated adaptively on all points. Therefore, instead of performing unpooling, we apply Gaussian sampling to obtain $n$ point features with a max value equal to $\mathbf{x}^P$. Suppose the input of $dec^P$ is $\hat{\mathbf{P}}^0$, then:

$$\hat{\mathbf{P}}_{i,k}^0 = min\left\{\mathcal{N}(0,1), \mathbf{x}_k^P\right\} \tag{2}$$

$\hat{\mathbf{P}}_{i,k}^0$ denotes the feature $k$ of point $i$, where $k = 1, 2, ..., 1024$ and $i = 1, 2, ..., n$. The sampled point features are then fed into three 1D transposed convolutional layers along the first dimension. We set kernel size to be 1, stride to be 1 and zero-padding. After three deconvolution layers, the final output of $dec^P$ is a $n \times F$ matrix consisting of $n$ points.

## III. Visualization and Results

Figure 2 shows the ROC curves and classification loss curves on both datasets. As could be discovered, our proposed MTUT method has higher Area Under the Curve (AUC) value compared with other baseline methods. In addition, classification loss of our proposed method achieves global minimum earlier than majority of other baseline architectures.

**Ablation Study.** The proposed MTUT method has two additional characteristics over traditional multi-view autoencoder: (1) face attribute vectors (Attr.); (2) adaptive embedding divergence (AED) loss. In this section, we will evaluate the contributions of these two additional methods on face recognition accuracy. Figure 3 present our ablation study varying model components. MTUT+AED+Attr. (the rightmost) represents the full model with both (1) and (2), whereas the others represent that we **exclude** one or both of the methods. As it could be discovered from Figure 3, AED has significant contribution over boosting classification accuracy on both datasets. The point could be even consolidated by that the accuracy of models without AED are even less than that of backbone architecture (eg. 92.95% vs. 94.87% on Kinect when testing on 2D). This proves that AED successfully optimizes the encoded embeddings in order to obtain robust performance on classification network. Although adding Attr. to face embeddings would theoretically guide reconstruction of face modalities, it leads to less obvious improvement on classification accuracy compared with AED.

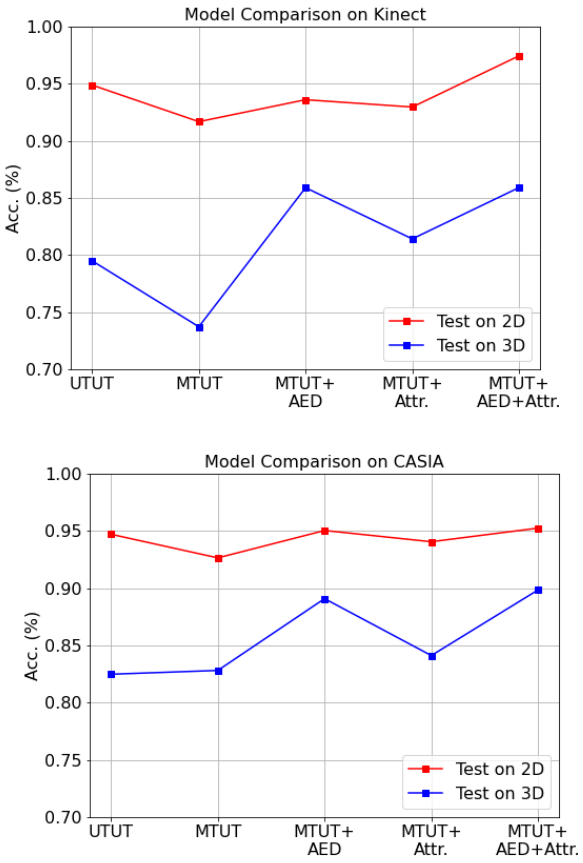


Fig. 3: Ablation study. Our methods are *MTUT+AED+Attr.* UTUT stands for unimodal training unimodal testing.

## IV. Related Works

**2D Face Recognition.** Compared with traditional face recognition methods, deep learning technique [8], [7], [6] employs larger number of layers to perform feature transformation and extraction, driving both accuracy and efficiency of face recognition up significantly. However, the performances are still restricted by pose and illumination variations. Compared with previous methods, our approach involves 3D data as a fundamental assistance to boost 2D model performance.

**3D Face Recognition.** 3D information of faces is crucial to face recognition modalities Our proposed approach is inspired by Qi *et al*. [5], which designs a novel neural net architecture called PointNet that is suitable for point cloud; it achieves permutation invariance by processing each point identically and independently with symmetrical function. Our proposed method applies the methodologies of PointNet to perform input spatial transformation and feature extraction process to obtain a high-level feature embedding.

**Multimodel Fusion.** Multimodal fusion helps effectively integrate information from a variety of modalities with the same goal of correct classification [2]. We perform multimodal fusion because multiple modalities have access to more comprehensive information necessary to combine for more robust performance. However, modalities like these are well-established when resources are complete and with high quality; especially when some modality views are unseen, the feature information of the seen views cannot be effectively transferred to unseen only through modal fusion.

**Co-Learning.** Previous problems could be mitigated with the technique of co-learning, which assists the modeling of modality with poor resources by exploiting embedded information of another modality [2]. We often perform co-learning when the assisting modality is used only during training. The proposed model is primarily inspired by [1] that effectively achieves Multimodal Training and Unimodal Testing (MTUT) by implementing Spatiotemporal Semantic Alignment (SSA) loss function. However, [1] failed to fuse the characteristics of different modalities and the results are largely decided by network performance on unimodal classification. Dumpala *et al*. [3] propose a cross-modal autoencoder (DCC-CAE) that maps the seen feature to unseen feature space through optimizing canonical correlation analysis cost function, but the power of their network would possibly be mitigated by the negative role played by any noisy modality.

## References


[1] M. Abavisani, H. R. V. Joze, and V. M. Patel. Improving the performance of unimodal dynamic hand-gesture recognition with multimodal training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1165–1174, 2019.

[2] T. Baltrušaitis, C. Ahuja, and L.-P. Morency. Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2):423–443, 2018.

[3] S. H. Dumpala, I. Sheikh, R. Chakraborty, and S. K. Kopparapu. Audio-visual fusion for sentiment classification using cross-modal autoencoder. NIPS, 2019.

[4] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[5] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[6] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[7] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In *Advances in neural information processing systems*, pages 1988–1996, 2014.

[8] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.