Object Centric Point Sets Feature Learning with Matrix Decomposition

Zijia Wang[1], Wenbin Yang[1], Zhisong Liu[1], Qiang Chen[1], Jiacheng Ni[1], and Zhen Jia[1]

Dell Technologies OCTO Research Office

Abstract. A representation matching the invariance/equivariance characteristics must be learnt to rebuild a morphable 3D model from a single picture input. However, present approaches for dealing with 3D point clouds depend heavily on a huge quantity of labeled data, while unsupervised methods need a large number of parameters. This is not productive. In the field of 3D morphable model building, the encoding of input photos has received minimal consideration. In this paper, we design a unique framework that strictly adheres to the permutation invariance of input points. Matrix Decomposition-based Invariant (MDI) learning is a system that offers a unified architecture for unsupervised invariant point set feature learning. The key concept behind our technique is to derive invariance and equivariance qualities for a point set via a simple but effective matrix decomposition. MDI is incredibly efficient and effective while being basic. Empirically, its performance is comparable to or even surpasses the state of the art. In addition, we present a framework for manipulating avatars based on CLIP and TBGAN, and the results indicate that our learnt features may help the model achieve better manipulation outcomes.

Keywords: object centric representation, 3D learning, diffusion model

1 Introduction

Understanding objects is one of the core problems of computer vision, especially in the avatar generation process, learning an object-centric representation is important in many downstream tasks [14].In machine learning, even a small attack [27] or image corruption [6] could produce a large accuracy decline. This is specifically important in 3D settings because a point cloud will be generated and can be varied based on different conditions. An object-centric representation is a graceful representation that can handle the distribution shift [4].

In this paper, we investigate deep learning architectures that are able to reason about three-dimensional geometric data, such as point clouds or meshes. In order to accomplish weight sharing and other kernel improvements, most convolutional architectures need extremely regular input data formats. Examples of such formats are picture grids and 3D voxels. Before feeding point clouds or meshes to a deep learning architecture, the majority of researchers often

convert such data to conventional 3D voxel grids or collections of pictures (for example, views) since point clouds and meshes do not have a regular format. This particular modification of the data representation, on the other hand, results in data that is needlessly large in volume, and it also introduces quantization artifacts, which have the potential to conceal the inherent invariances of the data.

To solve these problems, we propose a nearly non-parametric framework to learn a more useful representation and reconstruct the 3D avatar from the portraits. Concretely, our contributions to this paper are:

- A self-supervised learning framework to learn the canonical representation of the input. Two loss functions are introduced to make sure the learned representations are invariant/equivariant.
- Tensor decomposition for representation learning. The learned representation could satisfy the invariance/equivariance properties, which could be used in the 3D registration part which is the key component in 3D morphable model learning.
- A generalized application framework to deal with 3D images. We use the 3D morphable model reconstruction task as an example here, the results of the generated avatar show that our proposed algorithm can deal with the avatar manipulation well.

2 Related work

2.1 Point Cloud Features

Current point cloud feature extraction algorithms are mostly handcrafted based on one specific task [18]. These features contain certain statistical properties that are invariant to certain transformations. Therefore, they can be categorized into intrinsic (local features) [1, 2, 23] and extrinsic (global features) [3, 13, 16, 20, 21]. However, it's also necessary to optimally combine these properties. Although [18] tried to perform the trade-off to find the best feature combination, it's not trivial to make the whole process explicit and efficient.

2.2 Deep 3D representations

Currently, there exist many approaches for 3D feature learning like Volumetric CNNs [17, 19, 26] which utilize 3D convolutional neural networks to deal with voxelized shapes. However, data sparsity and computation cost of 3D convolution naturally limit the ability of the representations learned from these networks. Then FPNN [15] and Vote3D [25] proposed some metrics to solve the problem brought by data sparsity, but when these methods come to very large point clouds, their operations based on space volumes constrain them. These days, some new powerful methods like Multiview CNNs [22, 19] and Feature-based DNNs [7, 12] are proposed. However, the representative ability of the extracted features is still one of the key constraints of these metrics. Therefore, in this paper, we'll try to solve this problem.

3 Object-centric representation learning

The key innovation of the proposed method is to utilize the simple but powerful matrix decomposition to generate invariance/equivariance properties for points set. In the section3.2, we first introduce the matrix decomposition module in our method, then in section 3.3 we reveal our novel unsupervised pipeline. In the following, section 3.4, we develop the decoder for input point sets reconstruction. At last, we discuss the avatar generation extension for the proposed method in the section 3.6.

3.1 Problem definition

We design a deep learning framework that directly consumes unordered point sets as inputs. A point cloud is represented as a set of 3D points [18]

$$P = \{P_n | n = 1, ..., N\}, \tag{1}$$

where each point P_n is a vector of its (x, y, z) coordinate plus extra feature channels such as color, normal, etc. For simplicity and clarity, unless otherwise noted, we only use the (x, y, z) coordinate as our point's channels, as shown in fig. 1.

The input point set is a subset of points from Euclidean space. It has three main properties:[18]

- Unordered. Unlike pixel arrays in images or voxel arrays in volumetric grids, the point cloud is a set of points.
- Interaction among points. The points are from a space with a distance metric. It means that points are not isolated, and neighboring points form a meaningful subset. Therefore, the model needs to be able to capture local structures from nearby points, and the interactions among local structures.
- Invariance under transformations. As a geometric object, the learned representation of the point set should be invariant to certain transformations. For example, rotating and translating points altogether should not modify the global point cloud category or the segmentation of the points.

3.2 Matrix decomposition for invariant and equivariant feature learning

An overview of MD in the feature space is depicted in fig. 3. The point cloud input will be transferred to a feature matrix $X \in R^{M \times d}$, then we can decompose the matrix in the following manner:

$$X = UV + E \tag{2}$$

where $U \in R^{M \times k}$ and $V \in R^{k \times d}$ with $k < d$ are the decomposition factors; E is the residual. U can be considered as activation factor, which should be

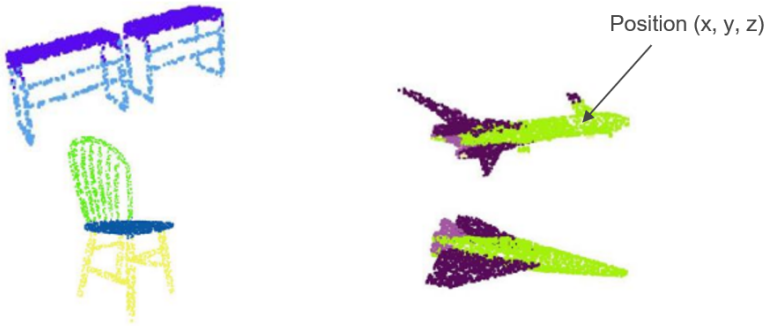

Fig. 1. Some examples of 3d point sets. Different colors indicate different semantic parts of an object.

invariant and capture the most important information for dataset. While V can be considered as the template factor, which is equivariant to each input data sample. Therefore, UV is the low rank approximation of X[11]. As shown in fig 3, the MD is a white-box unsupervised decomposition, which utilized the low-rank properties of the Feature Matrix. In MD, it can be conducted online[10].

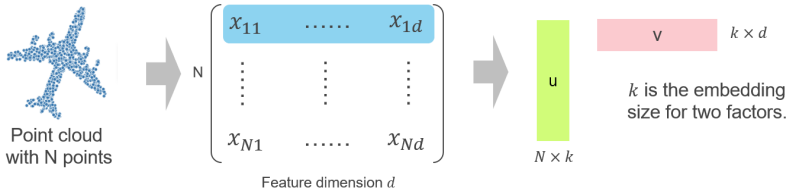

Fig. 2. The decomposition of the feature matrix. The invariant part U records the relative location, and the equivariant part V records the absolute location in world coordination.

3.3 Encoder

The encoder training pipeline of the proposed method is illustrated in the fig. 3. Our network is trained by feeding **pairs of randomly rotated copies of the same shape**. The input point clouds are randomly generated from two random transformations $T_1, T_2 \in R^{(3)}$ (rotating and translating). Note that we train such a decomposition in a fully unsupervised fashion and that the network only ever sees randomly rotated point clouds.

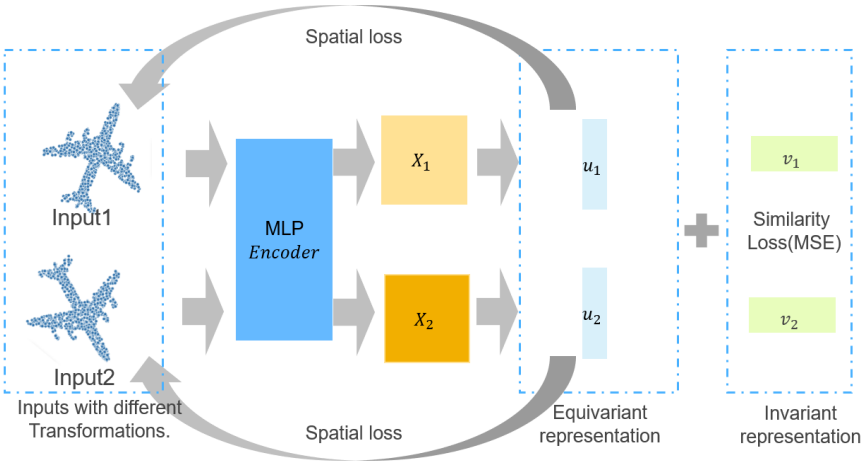

Fig. 3. The pipeline of the proposed method. The framework can take 2D- or 3D-points set as input.

As the two input point sets are of the same object, so the similarity loss can be expressed as

$$L_{sim} = \|V_1 - V_2\|_F \tag{3}$$

where $\|\cdot\|_F$ is the Frobenius norm of the matrix. Then, to learn the equivalence features. we ask for the network to learn a localized representation of the geometry. we define the following spatial loss for each input point set.

$$L_{spa}(P) = tr(U^T W U) \tag{4}$$

where $tr(\cdot)$ is the trace of the matrix; W the weight matrix of 3D points set. The $W(m,n)$ is the weight between two points, and can be calculated as:

$$W(m,n) = exp(\frac{\|P_m - P_n\|_2^2}{\sigma^2}) \tag{5}$$

where σ is the parameter to control the distance. For the sake of effectiveness, one can just use part of randomly selected points for reconstruction. The spatial loss considers the spatial relationship of the input 3D point set, as shown in the fig. 7.

After training, we will choose the 3D points set with the minimal l_1 norm of U as the reference point set.

3.4 Reconstruction Decoder

For downstream tasks:

– Classification: need the invariant representation.

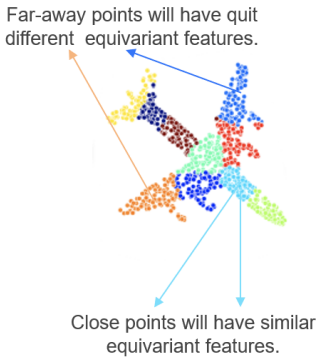

Fig. 4. The spatial loss.

– Segmentation: need the equivariant representation.
– Point sets reconstruction: need the reconstructed matrix UV and an addictive decoder that transforms the latent feature into 3D point sets.

It is clear that, in applications like Avatar[8], one should reconstruct the lantern features to a 3D point set. In this paper, we design a decoder shown in the fig. 5. The decoder MSE loss can be expressed as

$$L_{dec} = \frac{1}{N}\|P - \hat{P}\|_F \tag{6}$$

where $\hat{P}$ is the output matrix of the decoder.

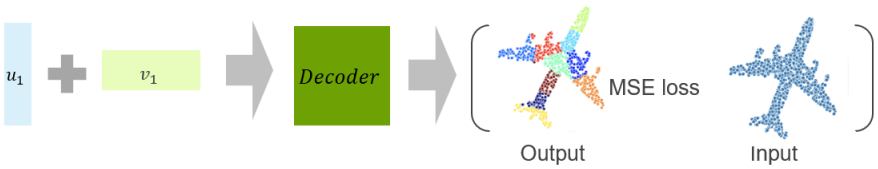

Fig. 5. The decoder for point set reconstruction.

The decoder can be trained with/without the framework proposed in the section3.3. When a train with the encoder3.3, the loss can be calculated as:

$$L = L_{sim} + \alpha \sum_{i}^{2} L_{spa}^{i} + \beta \sum_{i=1}^{2} L_{dec}^{i} \tag{7}$$

where α and β are weights to control the loss values. The joint training algorithm is summed in algorithm 1.

Algorithm 1: MDI Training

Input : Dataset $\{P\} \in \mathcal{P}$; Weight α and β, Training epoch number T
Output: Encoder and Decoder
Initialize: model parameters for Encoder and Decoder
for $t \leftarrow 1$ *to* T **do**
  for *each mini batch* B **do**
    for *each object* **do**
      compute inputs point randomly generated from two random transformations
      compute loss in eq.7
    compute the sum of all objects
    update parameters using back propagation
return

3.5 Theoretical analysis

In this section, we intend to illustrate why the low-rank assumption is beneficial for modeling the global context of representations by providing an example. The low-rank assumption is advantageous because it illustrates the inductive bias that low-level representations include fewer high-level ideas than the scale of the representations. Consider a picture of a person walking on the road. The route will be described by a large number of hyperpixels retrieved using a CNN's backbone. Notabene que la carretera puede ser considerada como repeticiones de pequeos fragmentos de carretera, por lo que se puede representar la carretera mediante la modelación y It is mathematically equal to locating a limited set of bases D corresponding to various road patches and a coefficient matrix C that records the relationship between the elementary road patches and the hyper-pixels. This example demonstrates that in an ideal setting, high-level notions, such as the global context, might be low-ranking. The hyper-pixels describing the road patches have semantic properties that are similar. Nevertheless, owing to the vanilla CNN's ineffectiveness un modeling long-distance relationships is reflected in its learnt representation, which includes too many local details and inaccurate information and lacks global direction. Imagine the subject in the photograph wearing gloves. When we see the gloves patch in our community, we assume it defines gloves. When the broader context is considered, it becomes clear that this patch is a portion of a person. The semantic information is hierarchical, depending on the amount of comprehension desired.

The objective of this section is to enable networks to comprehend the context globally by means of the low-rank recovery formulation. Incorrect information, notably redundancies and incompletions, are modeled as a noise matrix. To highlight the global context, we split the representations into two parts, a low-rank global information matrix and a local equivariant matrix, using an optimization approach to recover the clean signal subspace, eliminate the noises, and improve the global information through the skip connection. The data might reveal how much global knowledge the networks need for a certain operation.

3.6 Measurement of learned features on avatar generation

Avatar generation can naturally be a good application for the testing of invariance and equivariance features. Fig. 6 shows the framework of our avatar generation, which is based on the TBGAN proposed in [9]. Given a pre-trained TBGAN generator $\mathcal{G}$, let $z \in \mathcal{R}^d$ denote a d-dimensional random input vector sampled from a Gaussian distribution $\mathcal{N}\left(0, \sigma^2\right)$ and e originally denote a one-hot encoded facial expression vector initialized to zero to obtain a neutral expression. However, in this paper, the equivariant feature derived from the generated image from CLIP becomes e. Let $\mathbf{c} \in \mathcal{C}$ denote an intermediate layer vector obtained by partial forward propagation of $\mathbf{z}$ and e through the generator $\mathcal{G}$. Our method first generates a textured mesh by using the generated shape, normal, and texture UV maps via cylindrical projection. Then given a text prompt t such as 'happy human', $\mathbf{c}$ is optimized via gradient descent to find a direction $\Delta\mathbf{c}$, where $\mathcal{G}(\mathbf{c} + \Delta\mathbf{c})$ produces a manipulated textured mesh in which the target attribute specified by t is present or enhanced, while other attributes remain largely unaffected. In our work, we optimize the original intermediate latent vector $\mathbf{c}$ using gradient descent and work in the 4×4 dense layer s of the TBGAN generator.

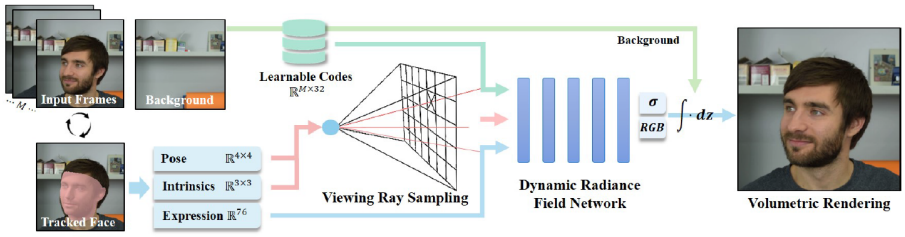

Fig. 6. The avatar generation framework for testing the learned feature extractor.

The optimized latent vector $c + \Delta c$ can then be fed into TBGAN to generate shape, normal, and texture UV maps, and finally a manipulated mesh with the target attributes. To perform meaningful manipulation of meshes without creating artifacts or changing irrelevant attributes, we use a combination of an equivariance loss, an identity loss, and an L2 loss as follows:

$$\arg\min_{\Delta\mathbf{c}\in\mathcal{C}} \quad \mathcal{L}_{\text{eq}} + \lambda_{\text{ID}}\mathcal{L}_{\text{ID}} + \lambda_{\text{L2}}\mathcal{L}_{\text{L2}} \tag{8}$$

where λ_{ID} and λ_{L2} are the hyperparameters of $\mathcal{L}_{ID}$ and $\mathcal{L}_{\text{L2}}$, respectively. While equivariance loss ensures that the user-specified attribute is present or enhanced, ID-loss and L2-loss leave other attributes unchanged, forcing disentangled changes. The identity loss $\mathcal{L}_{ID}$ minimizes the distance between the identity of the original renders and the manipulated renders:

$$\mathcal{L}_{\text{ID}} = \|(\mathbf{U}_{\mathbf{ori}} - \mathbf{U}_{\mathbf{edi}})\|_2 \tag{9}$$

where $\mathbf{U_{ori}}$ is the invariant feature of the original image and the $\mathbf{U_{edi}}$ is the invariant feature for edited image. Similarly, the equivariance loss $\mathcal{L}_{eq}$ can be defined as:

$$\mathcal{L}_{\text{eq}} = \|(\mathbf{V_{ori}} - \mathbf{V_{edi}})\|_2, \tag{10}$$

where $\mathbf{V_{ori}}$ and $\mathbf{V_{edi}}$ are the equivariant features of the original image and the edited image respectively. Finally, the L2 loss is used to prevent artifact generation and defined as:

$$\mathcal{L}_{\text{L.2}} = \|\mathbf{c} - (\mathbf{c} + \Delta\mathbf{c})\|_2 \tag{11}$$

For TBGAN and renderer, we follow the same settings in [9].

4 Experiment results

This section evaluates the proposed approach and compares it against State Of The Art methods. To evaluate our method, we rely on the ShapeNet (Core) dataset . We follow the category choices from AtlasNetV2 , using the airplane and chair classes for single category experiments, while for multi-category experiments we use all the 13 classes in ShapeNet (Core) dataset. Unless noted otherwise, we randomly sample 1024 points from the object surface for each shape to create our 3D point clouds.

For all our experiments we use the Adam optimizer with an initial learning rate of 0.001 and decay rate of 0.1. Unless stated otherwise, we use k=10 and feature dimension d=128.

Our network architecture:

- **Encoder.** Our architecture is based on the one suggested in[24]: a point net-like architecture with residual connections and attentive context normalization.
- **Decoder.** Our decoder architecture is similar to AtlasNetV2[5](with trainable grids).

The last section in this part shows the avatar reconstruction results based on the framework shown in the section 3.6.

4.1 Reconstruction Result

We evaluate the performance of our method for reconstruction against two baselines:

- AtlasNetV2[5], the State Of The Art auto-encoder which utilizes a multi-head patch-based decoder;
- 3D-PointCapsNet[28], an auto-encoder for 3D point clouds that utilize a capsule architecture.

As shown in the table.1 We achieve State Of The Art performance in both the aligned and unaligned settings.

We illustrate the reconstruction of 3D point clouds for all the methods. our method can provide semantically consistent decomposition, for example, the wings of the airplane have consistent colors.

	Aligned			Unaligned		
	Airplane	Chair	Multi	Airplane	Chair	Multi
3D-PointCapsNet	1.94	3.30	2.49	5.58	7.57	4.66
AtlasNetV2	1.28	2.36	2.14	2.80	3.98	3.08
MDI(Ours)	**0.93**	**2.01**	**1.66**	**1.05**	**3.75**	**2.20**

Table 1. Reconstruction – Performance in terms of Chamfer distance.

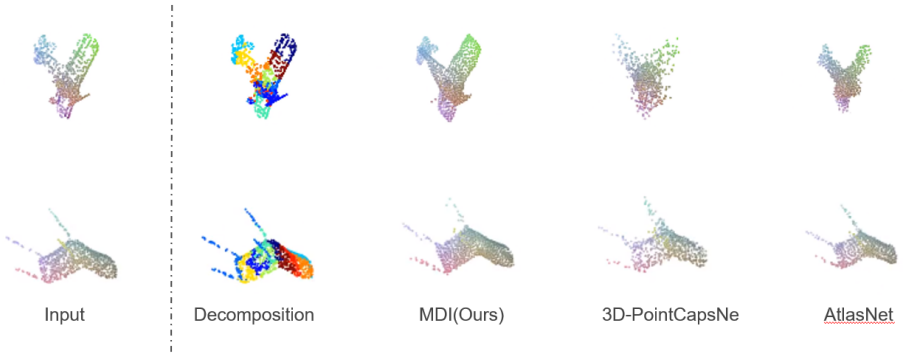

Fig. 7. Reconstruction results.

4.2 Classification Result

We compute the features from the auto-encoding methods compared in Section 4.1 – AtlasNetV2[5], 3D-PointCapsNet[28], and our learned invariance features. We use them to perform 13-way classification with Support Vector Machine (SVM) and K-Means clustering. Our results are superior to the other SOTA method. We argue that the joint invariant and equivariant feature learning with MD is important to unsupervised learning. This is especially obvious for the unaligned part because of the advantages of our learned invariant and equivariant features. And for aligned ones, we can also achieve competitive results.

	Aligned		**Unaligned**	
	SVM	K-Means	SVM	K-Means
AtlasNetV2	94.07	61.66	71.13	14.59
3D-PointCapsNet	93.81	65.87	64.85	17.12
MDI(Ours)	**93.78**	**71.42**	**86.58**	**49.93**

Table 2. Classification – Top-1 accuracy (%)

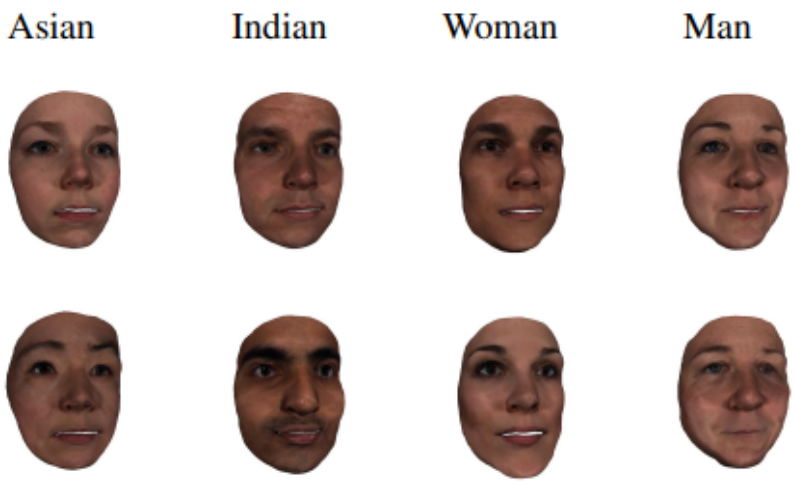

Fig. 8. Results of manipulation on equivariant features.

4.3 Avatar Generation

In this section, to show the power of our learned features, we follow the framework shown in the figure 6. By doing this, our method can be used to change their facial expressions such as 'smiling', 'angry', and 'surprised'. As can be seen in fig. 8, our method can successfully manipulate a variety of complex emotions on various input meshes with almost no change to other attributes. By comparing with the results of TBGAN [9], the advantages of directly manipulating equivariant features are obvious.

By manipulating the invariant features, we slightly change the framework shown in fig. 6. Concretely, the CLIP is used to generate the pictures with global features (invariant features), then the U is extracted from generated image. Then V is extracted from the original avatar images. The results also show that our method provides a global semantic understanding of more complex attributes such as 'man', 'woman', 'Asian', and 'Indian'. Fig. 9 shows the results for manipulations on various randomly generated outputs, where we can see that our method can perform complex edits such as ethnicity and gender.

5 Conclusion

In this paper, we design a novel Matrix Decomposition-based Invariant (MDI) learning framework, which can provide a unified architecture for unsupervised invariant point sets feature learning.

Though simple, MDI is highly efficient and effective. Empirically, it shows strong performance on point sets reconstruction and unsupervised classification.

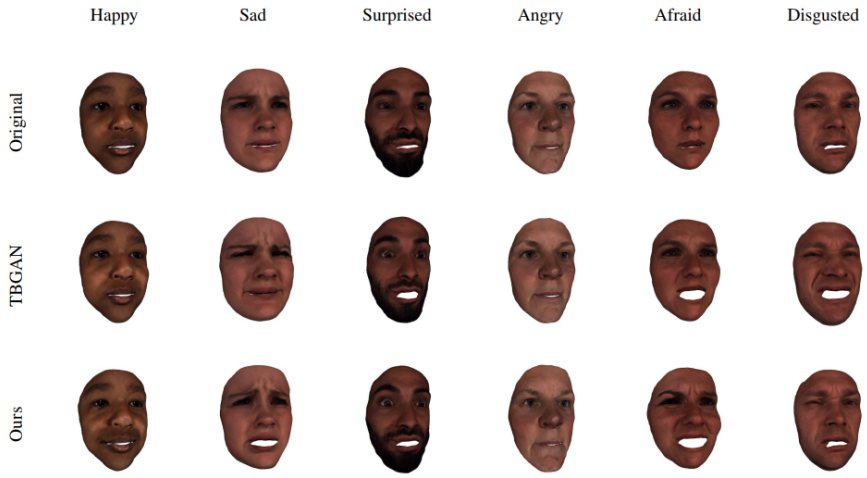

Fig. 9. Results of manipulation on invariant features.

Moreover, our framework will benefit other downstream like collaborative computing in avatar generation.

References

1. Aubry, M., Schlickewei, U., Cremers, D.: The wave kernel signature: A quantum mechanical approach to shape analysis. In: 2011 IEEE international conference on computer vision workshops (ICCV workshops). pp. 1626–1633. IEEE (2011)
2. Bronstein, M.M., Kokkinos, I.: Scale-invariant heat kernel signatures for non-rigid shape recognition. In: 2010 IEEE computer society conference on computer vision and pattern recognition. pp. 1704–1711. IEEE (2010)
3. Chen, D.Y., Tian, X.P., Shen, Y.T., Ouhyoung, M.: On visual similarity based 3d model retrieval. In: Computer graphics forum. vol. 22, pp. 223–232. Wiley Online Library (2003)
4. Creager, E., Jacobsen, J., Zemel, R.S.: Exchanging lessons between algorithmic fairness and domain generalization. CoRR **abs/2010.07249** (2020), https://arxiv.org/abs/2010.07249
5. Deprelle, T., Groueix, T., Fisher, M., Kim, V.G., Russell, B.C., Aubry, M.: Learning elementary structures for 3d shape generation and matching. arXiv preprint arXiv:1908.04725 (2019)
6. Duchi, J., Glynn, P., Namkoong, H.: Statistics of robust optimization: A generalized empirical likelihood approach. arXiv preprint arXiv:1610.03425 (2016)
7. Fang, Y., Xie, J., Dai, G., Wang, M., Zhu, F., Xu, T., Wong, E.: 3d deep shape descriptor. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2319–2328 (2015)
8. Gafni, G., Thies, J., Zollhofer, M., Nießner, M.: Dynamic neural radiance fields for monocular 4d facial avatar reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8649–8658 (2021)

9. Gecer, B., Lattas, A., Ploumpis, S., Deng, J., Papaioannou, A., Moschoglou, S., Zafeiriou, S.: Synthesizing coupled 3d face modalities by trunk-branch generative adversarial networks. In: European conference on computer vision. pp. 415–433. Springer (2020)
10. Geng, Z., Guo, M.H., Chen, H., Li, X., Wei, K., Lin, Z.: Is attention better than matrix decomposition? arXiv preprint arXiv:2109.04553 (2021)
11. Guan, N., Tao, D., Luo, Z., Shawe-Taylor, J.: Mahnmf: Manhattan non-negative matrix factorization. arXiv preprint arXiv:1207.3438 (2012)
12. Guo, K., Zou, D., Chen, X.: 3d mesh labeling via deep convolutional neural networks. ACM Transactions on Graphics (TOG) **35**(1), 1–12 (2015)
13. Johnson, A.E., Hebert, M.: Using spin images for efficient object recognition in cluttered 3d scenes. IEEE Transactions on pattern analysis and machine intelligence **21**(5), 433–449 (1999)
14. Li, N., Raza, M.A., Hu, W., Sun, Z., Fisher, R.: Object-centric representation learning with generative spatial-temporal factorization. Advances in Neural Information Processing Systems **34** (2021)
15. Li, Y., Pirk, S., Su, H., Qi, C.R., Guibas, L.J.: Fpnn: Field probing neural networks for 3d data. Advances in neural information processing systems **29** (2016)
16. Ling, H., Jacobs, D.W.: Shape classification using the inner-distance. IEEE transactions on pattern analysis and machine intelligence **29**(2), 286–299 (2007)
17. Maturana, D., Scherer, S.: Voxnet: A 3d convolutional neural network for real-time object recognition. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 922–928. IEEE (2015)
18. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 652–660 (2017)
19. Qi, C.R., Su, H., Nießner, M., Dai, A., Yan, M., Guibas, L.J.: Volumetric and multi-view cnns for object classification on 3d data. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5648–5656 (2016)
20. Rusu, R.B., Blodow, N., Beetz, M.: Fast point feature histograms (fpfh) for 3d registration. In: 2009 IEEE international conference on robotics and automation. pp. 3212–3217. IEEE (2009)
21. Rusu, R.B., Blodow, N., Marton, Z.C., Beetz, M.: Aligning point cloud views using persistent feature histograms. In: 2008 IEEE/RSJ international conference on intelligent robots and systems. pp. 3384–3391. IEEE (2008)
22. Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E.: Multi-view convolutional neural networks for 3d shape recognition. In: Proceedings of the IEEE international conference on computer vision. pp. 945–953 (2015)
23. Sun, J., Ovsjanikov, M., Guibas, L.: A concise and provably informative multi-scale signature based on heat diffusion. In: Computer graphics forum. vol. 28, pp. 1383–1392. Wiley Online Library (2009)
24. Sun, W., Tagliasacchi, A., Deng, B., Sabour, S., Yazdani, S., Hinton, G., Yi, K.M.: Canonical capsules: Self-supervised capsules in canonical pose. In: Thirty-Fifth Conference on Neural Information Processing Systems (2021)
25. Wang, D.Z., Posner, I.: Voting for voting in online point cloud object detection. In: Robotics: Science and Systems. vol. 1, pp. 10–15. Rome, Italy (2015)
26. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3d shapenets: A deep representation for volumetric shapes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1912–1920 (2015)

27. Zhang, B.H., Lemoine, B., Mitchell, M.: Mitigating unwanted biases with adversarial learning. In: Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society. pp. 335–340 (2018)
28. Zhao, Y., Birdal, T., Deng, H., Tombari, F.: 3d point capsule networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1009–1018 (2019)