Supplementary Materials for ECCV 2024 paper SA-DVAE: Improving Zero-Shot Skeleton-Based Action Recognition by Disentangled Variational Autoencoders

Sheng-Wei Li[1], Zi-Xiang Wei[2], Wei-Jie Chen[2], Yi-Hsin Yu[2], Chih-Yuan Yang[3,4], and Jane Yung-jen Hsu[2,3]

[1] Graduate Institute of Networking and Multimedia, National Taiwan University
[2] Department of Computer Science and Information Engineering, National Taiwan University
[3] Department of Artificial Intelligence, Chang Gung University
[4] Artificial Intelligence Research Center, Chang Gung University
{r11944004,r12922147,r12922051,r12922220,yjhsu}@csie.ntu.edu.tw,
cyyang@cgu.edu.tw

A Hyperparameter Search Space and Sensitivity

We show our search space and initial values in Tab. A.

Table A: Hyperparameter search space and initial values

No.	Hyperparameter	Space	Initial
1	β_x and β_y	(0, 1.0)	
2	Learning rate's exponent	(-6.0, -3.0)	-5
3	Batch size	{32, 64, 128, 256}	64
4	Discriminator steps n_d	{1, 2, 3, ..., 16}	10
5	Hidden dim. of z_x^r and z_y	{128, 144, 160, ..., 256}	192
6	Hidden dim. of z_x^v	{8, 12, 16, ..., 32}	8

We first fix No. 2∼6 and randomly sample No. 1 in uniform distribution 5 times. We choose the one generating the highest GZSL harmonic mean on the validation set. Then we fix No. 1 and randomly sample No. 2∼6 100 times.

Table B shows the influence of β_x and β_y on the experiments of Tables 6 and 7 in the main paper. As reported in Table 5 in the main paper, we use β_x as 0.023 and β_y as 0.011 because they perform best on the validation set. We leave out β_x and $\beta_y \geq 0.2$ because their performance is low.

B Feature Extractors

We show an example by re-organizing Tables 6 and 7 in the main paper as Tab. C. Their dataset, splits, and hyperparameters are the same and the only

Table B: Sensitivity of β_x and β_y on ZSL and GZSL metrics.

β_x	β_y	ZSL	GZSL		
		Acc	Acc_s	Acc_u	*H*
0.010	0.010	83.60	81.14	67.14	73.48
0.050	0.010	**84.39**	73.87	73.24	73.55
0.010	0.050	83.13	77.61	71.22	74.28
0.050	0.050	83.62	70.74	74.23	72.44
0.100	0.100	82.79	75.96	68.86	72.24
0.200	0.200	76.12	71.79	62.90	67.05
0.023	0.011	84.20	78.16	72.60	**75.27**

difference lies in feature extractors. Experimental results show that extractors matter and our proposed ST-GCN+CLIP works best.

Table C: Average ZSL accuracy and GZSL metrics (%) of different feature extractors under the random split setting on NTU-60.

Feature Extractors	ZSL	GZSL		
	Acc	Acc_s	Acc_u	*H*
ST-GCN [5] + Sentence-BERT [4]	74.38	71.39	61.02	65.80
PoseC3D [1] + CLIP [3]	81.84	83.48	66.89	74.27
ST-GCN [5] + CLIP [3]	**84.20**	78.16	72.60	**75.27**

C Combining with Existing Methods

To potentialy improve our performance, we combine our method with pose canonicalization on skeleton data [2] and enhanced class descriptions by a large language model proposed in SMIE [6]. We will discuss the details and experimental results in the following sections.

C.1 Pose Canonicalization on Skeleton Data

The difference in the forward direction of the skeleton data introduces additional noise into the training process. Therefore, we implement the method proposed by Holden *et al*. [2] to canonicalize the skeleton data by rotating them so that they face the same direction. We compute the cross product between the vertical axis and the average vector of the left and right shoulders and hips to determine the new forward direction of the body. We then apply a rotation matrix to canonicalize the pose.

Tables D and E present the experimental results under random split settings listed in Table 5 of the main paper. In zero-shot settings, we observe that canonicalization of skeleton data has little effect on model performance. For generalized zero-shot settings, we note a slight decrease in both seen and unseen accuracies. We hypothesize that this is because canonicalization reduces the variation in the skeleton dataset. This reduction in diversity limits the range of examples the model encounters during training, which may ultimately impair its ability to generalize effectively.

Table D: Average ZSL accuracy (%) under the random split setting on the NTU-60, NTU-120, and PKU-MMD datasets.

Method	NTU-60 55/5 split	NTU-120 110/10 split	PKU-MMD 46/5 split
SA-DVAE	**84.20**	**50.67**	66.54
SA-DVAE + pose canonicalization	84.03	50.04	**67.56**

Table E: Average GZSL metrics: seen class accuracy Acc_s, unseen class accuracy Acc_u, and their harmonic mean H (%) under the random split setting on the NTU-60, NTU-120, and PKU-MMD datasets.

Method	NTU-60 55/5 splits			NTU-120 110/10 split			PKU-MMD 46/5 split		
	Acc_s	Acc_u	H	Acc_s	Acc_u	H	Acc_s	Acc_u	H
SA-DVAE	78.16	72.60	**75.27**	58.09	40.23	**47.54**	58.49	51.40	**54.72**
SA-DVAE + pose canonicalization	72.84	69.85	71.31	56.78	35.22	43.47	54.13	50.60	52.30

C.2 Enhanced Class Descriptions by a Large Language Model (LLM)

Zhou *et al*. [6] propose to use an LLM to augment class descriptions with richer action-related information and we directly compare our and their methods by using their augmented descriptions. We report results using the same setting for random split and list our hyperparameters in Table F, and generate results shown in Tables G and H, which show that SA-DAVE outperforms SMIE using augmented descriptions in both ZSL and GZSL protocols and LLM-augmented descriptions significantly improve unseen accuracy while marginally decreasing seen accuracy. This is consistent with the pattern observed in the ablation study, indicating that the models achieve a more balanced prediction with minimal bias toward seen or unseen classes.

Table F: Settings for LLM-augmented class descriptions under the random split setting.

	NTU-60	NTU-120
Skeleton Feature Extractor	ST-GCN [5]	
Text Feature Extractor	CLIP-ViT-B/32 [3]	
Epochs	10	
Optimizer	Adam	
No. of unseen classes	5	10
Optimizer Momentum	$\beta_1 = 0.9, \beta_2 = 0.999$	
Batch size	32	24
Learning rate	4.94e-05	2.13e-05
Weights of D_{KL} in $\mathcal{L}_{VAE}$	$\beta_x = 0.023, \beta_y = 0.011$	
Weight of $\mathcal{L}_T$	$\lambda_2 = 0.011$	
Discriminator steps n_d	4	16
Hidden dim. of z_x^r and z_y	96	304
Hidden dim. of z_x^v	8	12

Table G: ZSL accuracy (%) with LLM-augmented class descriptions on the NTU-60 and NTU-120 datasets.

Method	NTU-60 55/5 split	NTU-120 110/10 split
SMIE [6]	65.08	46.40
SMIE + augmented text [6]	70.89	52.04
SA-DVAE	84.20	50.67
SA-DVAE + augmented text	**87.61**	**57.16**

Table H: GZSL metrics (%) with LLM-augmented class descriptions on the NTU-60 and NTU-120 datasets.

Method	NTU-60 55/5 splits			NTU-120 110/10 split		
	Acc_s	Acc_u	H	Acc_s	Acc_u	H
SA-DVAE	78.16	72.60	75.27	58.09	40.23	47.54
SA-DVAE + augmented text	74.54	76.50	**75.51**	53.32	48.36	**50.72**

References

1. Duan, H., Zhao, Y., Chen, K., Lin, D., Dai, B.: Revisiting skeleton-based action recognition. In: CVPR. pp. 2969–2978 (2022)
2. Holden, D., Saito, J., Komura, T.: A deep learning framework for character motion synthesis and editing. ACM Transactions on Graphics (TOG) **35**(4), 1–11 (2016)
3. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
4. Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084 (2019)
5. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: AAAI. vol. 32 (2018)
6. Zhou, Y., Qiang, W., Rao, A., Lin, N., Su, B., Wang, J.: Zero-shot skeleton-based action recognition via mutual information estimation and maximization. In: ACM MM (2023)