Learning Shared Embedding Representation of Motion and Text Using Contrastive Learning

Junpei Horie[1], Wataru Noguchi[2], Hiroyuki Iizuka[2,3], and Masahito Yamamoto[2,3]

[1] Graduate School of Information Science and Technology, Hokkaido University, Japan
[2] Faculty of Information Science and Technology, Hokkaido University, Japan
[3] Center for Human Nature, Artificial Intelligence, and Neuroscience, Hokkaido University, Japan
(Tel: +81-11-706-6445)

horie@ist.hokudai.ac.jp

Abstract: Multimodal learning of motion and text tries to find the correspondence between skeletal time-series data acquired by motion capture and the text that describes the motion. In this field, good associations can realize both motion-to-text and text-to-motion applications. However, the previous methods failed to associate motion with text, taking into account details of descriptions, for example, whether to move the left or right arm. In this paper, we propose a motion-text contrastive learning method for making correspondences between motion and text in a shared embedding space. We showed that our model outperforms the previous studies in the task of action recognition. We also qualitatively show that, by using a pre-trained text encoder, our model can perform motion retrieval with detailed correspondences between motion and text.

Keywords: multi-modal learning, contrastive learning, skeleton-based action recognition, motion retrieval

1 INTRODUCTION

Humans process sensory information by integrating multiple modalities, including vision, hearing, smell, touch, and taste. In the research field of machine learning, the way to effectively utilize such multimodal data in an integrated manner has been developed as multimodal learning. Multimodal learning is a method of learning how to process information from multiple modalities so that tasks that are difficult to accomplish in a single modality can be performed.

Multimodal learning of motion and text tries to find the correspondence between skeletal time-series data acquired by motion capture and the text that describes the motion. In this field, establishing good associations between motion and text can realize both motion-to-text and text-to-motion applications. To achieve a multimodal understanding of motion and text, motion and text must be associated with understandings of their linguistic concepts and structures.

One of the multimodal learning tasks for motion and text is text-to-motion generation. Text-to-motion generation is the task of generating motions that reflect the textual representation of the input text. To generate motion that matches the textual representation, the model needs to learn a shared embedding of motion and text, and various methods have been studied [1, 2]. However, conventional text-to-motion generation methods learn a one-to-one correspondence between motion and text. Therefore, it is difficult to convert unseen text into an appropriate embedding representation, and motion could not be generated from an unlearned text representation.

To overcome the problem of generalization for unseen text, some previous works incorporated pre-trained text representations. Tevet et al. used CLIP [3, 4] trained with a large amount of image-text pair data and associated CLIP's high-quality text embedding representation with the motion embedding representation [5]. As a result, they could generate motion that reflected the textual representation by predicting the embedding of unseen text from the CLIP embedding representation, even for text with various representations that were not included in the training data. However, the authors noted that it is difficult to generate motions considering modifiers, for example, to determine which arm is moved. It may be because the textual representation in CLIP was obtained by learning images and texts, not by learning the correspondence between motions and texts. Therefore, it may be necessary to learn the correspondence between the textual representations of CLIP and motions that take detailed textual representation into account.

In this paper, we propose a motion-text contrastive learning [6, 7, 8, 9] method for learning shared embedding representations of motion and text. In our model, we use CLIP's pre-trained text encoder, but different from the previous studies, we additionally learn the correspondence between motion and text through contrastive learning. We trained our model on motion-text data, pairs of skeleton data captured by motion capture, and text describing the motion. Then, we evaluated our model on action recognition and motion retrieval tasks. In addition, we qualitatively evaluated the abilities in motion retrieval to recognize motion components, such as right and

left, and to infer correspondence between untrained motion and text.

2 METHOD

The proposed model is shown in Fig. 1. The model consists of encoders for both modalities of motion and text. The encoders are trained by contrastive learning for obtaining a shared embedding space of motion and text. The contrastive learning is performed on sets of motion and text pairs. Given a batch of n pairs of motion and text, each motion and text is converted into a shared embedding by the encoders, respectively. The encoders are optimized so that the similarity between paired embeddings becomes high and the non-paired embedding motion and text become low. After the optimization, the shared embeddings can be used for action recognition and motion retrieval based on the similarities between embeddings.

2.1 Motion Encoder

The motion encoder consists of a graph convolutional network [10]. The skeleton of the input motion data is represented by a graph whose nodes are joints of a human skeleton. The graph is then convolved in the temporal and spatial directions for recognizing both the temporal and spatial context of input motions [11]. An input fixed length motion is converted to 512-dimensional embedding.

2.2 Text Encoder

We used CLIP's text encoder [3] trained with image-text pair data collected from the internet as the pre-trained model. The text encoder consists of the transformer network, a model for natural language processing with an attention mechanism [12, 13]. It processes a series of words in the text and outputs 512-dimensional distributed embeddings for each word, including [SOS] and [EOS] tokens which indicate the beginning and end of the sentence. The representation corresponding to the [EOS] token is used as the text embeddings for contrastive learning.

2.3 Similarity in Shared Embedding Space

The motion and text input pairs are converted into embeddings by encoders. By calculating similarities for each possible combination between the batch of n motion and text pairs, we obtain the similarity scores with size $n \times n$. The similarity score is calculated as cosine similarities between the embeddings of motion and text as follows:

$$T_{\mathrm{f}} = \mathrm{TextEncoder}(T) \tag{1}$$

$$M_{\mathrm{f}} = \mathrm{MotionEncoder}(M) \tag{2}$$

$$T_{\mathrm{e}} = \frac{T_{\mathrm{f}}}{\|T_{\mathrm{f}}\|} \tag{3}$$

$$M_{\mathrm{e}} = \frac{M_{\mathrm{f}}}{\|M_{\mathrm{f}}\|} \tag{4}$$

$$logits = T_{\mathrm{e}} \cdot M_{\mathrm{e}}^{\top} \tag{5}$$

where M and T are batch inputs of motion and text, respectively.

2.4 Loss Function

The encoders are optimized so that the similarity between the embeddings of paired motion and text becomes high and those of non-paired motion and text become low. The loss function is designed such that the diagonal elements, which correspond to similarities of the paired motion and text, become close to 1 and the other elements become close to 0 [14, 15]. First, we calculate the cross-entropy errors in the row direction of $logits$ to obtain the $loss_{\mathrm{t}}$. Text loss $loss_{\mathrm{t}}$ is calculated as the average of cross entropies between $logits$ and target t for each row.

$$t = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{pmatrix} \tag{6}$$

$$y_{i,j} = \frac{\exp(logits_{[i,j]})}{\sum_{i'=1}^{n} \exp(logits_{[i',j]})} \tag{7}$$

$$loss_{\mathrm{t}} = -\frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{n} t_{i,j} \log y_{i,j} \tag{8}$$

Second, we calculate the cross-entropy errors in the column direction of $logits$ to obtain the $loss_{\mathrm{m}}$. Motion loss $loss_{\mathrm{m}}$ is calculated as the average of cross entropies between $logits$ and target t for each column.

$$y_{j,i} = \frac{\exp(logits_{[j,i]})}{\sum_{i'=1}^{n} \exp(logits_{[j,i']})} \tag{9}$$

$$loss_{\mathrm{m}} = -\frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{n} t_{j,i} \log y_{j,i} \tag{10}$$

Finally, the average of $loss_{\mathrm{m}}$ and $loss_{\mathrm{t}}$ is taken as the final loss.

$$loss = \frac{(loss_{\mathrm{t}} + loss_{\mathrm{m}})}{2} \tag{11}$$

3 EXPERIMENTS

3.1 Setting

This study used the BABEL dataset [16], in which motions are paired with text describing the motion. The BABEL dataset consists of 40 hours of mocap data, and the mocap data consists of skeleton data with 21 joints [17, 18]. In this study, motion data were downsampled to 30 fps, and the length of the motion was set to 150 frames which corresponds to five seconds because it is possible to identify what kind of movement is enough in five seconds. Motions of less than five seconds were repeated to be five seconds, and motions

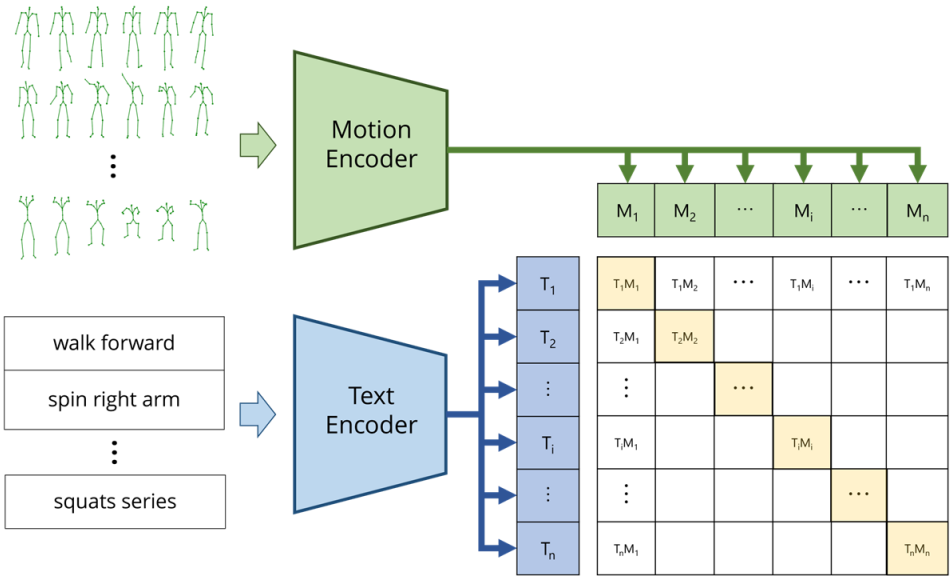


Fig. 1: Overview of our model. Motion-text input pair data are converted to embeddings by the encoders, and shared embedding representations of motion and text are learned by contrastive learning.

of more than five seconds were split every five seconds. Each motion is assigned an annotation describing the motion and 120 different action categories. There are recognition tasks to recognize the action category of motion for class 60 (BABEL-60) and class 120 (BABEL-120). The 48,978 data in the training set were used to train the model, and the 18,368 data in the validation set were used for evaluation by action recognition and motion retrieval. The example of the data set, with skeleton data every 0.1 seconds, is shown in Fig. 2.

The models were implemented by using PyTorch's deep learning framework. In this study, we used CTRGCN [19] as a motion encoder for the baseline model. We also used 2sAGCN [20] to compare our results with those of previous studies. The two models share the same structure. There are composed of 10 blocks of graph convolution layers, and the number of channels in each block is 64-64-64-64-128-128-128-256-256-256. The temporal dimension is halved on the 5th and 8th blocks by stride operation. After the graph convolution block, the embedding is transformed into 512-dimensional embeddings by 2-layer MLP. The GELU [21] function was used as the activation function for the first layer MLP. The output of the motion encoder is the layer-normalized motion embedding [22, 23]. The transformer of the text encoder used in this study was configured with 12 layers, 512 width, and 8 attention heads. In this study, we used five models to evaluate our model: 2sAGCN(CE), 2sAGCN-C(ours), CTRGCN(CE), CTRGCN-T(ours), and CTRGCN-C(ours). The model trained using only cross-entropy is denoted "(CE)". In this model, only the motion encoder is used, not the text encoder. The motion encoder is trained by computing cross-entropy with the motion encoder's output and the ground-truth action category. The model trained using the contrastive loss of our method is denoted "(ours)". The text encoder with no pre-trained is denoted "-T" and the pre-trained text encoder of CLIP is denoted "-C". The models were optimized by the SGD algorithm for 300 epochs with a momentum of 0.9, a learning rate of 0.0001, a weight decay of 0.0001, and a batch size of 64. Below, we evaluate the performances in action recognition and motion retrieval by the trained model.

3.2 Skelton Based Action Recognition

First, we performed the 60-class and 120-class recognition to test the model's ability of action recognition. We trained our model using the text of the action category assigned to the motion as input to the text encoder. In this way, we can evaluate the effect of contrastive learning independent of the complexity of the text. The average of the results of five training runs for each model is shown in Tab. 1. Top-1 is the percentage of the highest predicted category that matches the ground-truth category, Top-5 is the percentage of the predictions where the ground truth category is among the top 5 predicted categories, and Top-1-norm is the average of Top-1 values across categories.

The CTRGCN-C(ours) model using CLIP's text encoder was more accurate than the CTRGCN-T(ours) model using the no pre-trained transformer. For this result, pre-trained text encoders were found to be effective in performing action recognition tasks with our method.

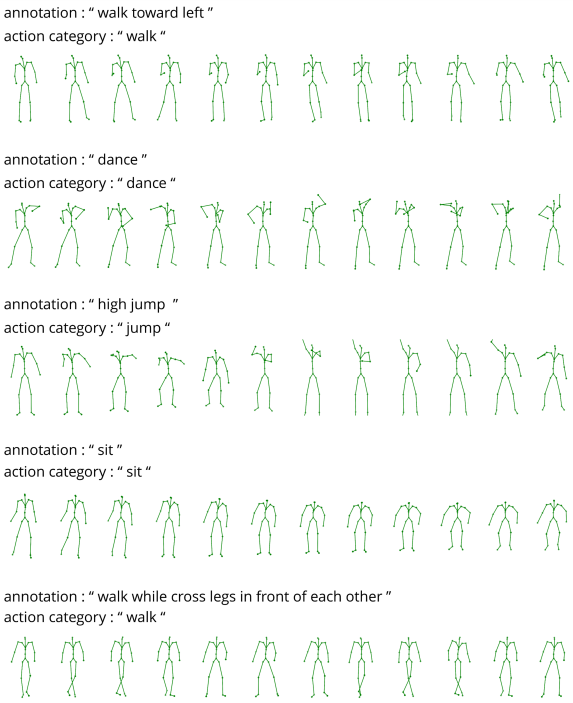


Fig. 2: Examples of BABEL dataset [16]. The skeleton consists of 21 joints, and each motion is assigned an annotation that describes the motion and the action category that represents the type of motion.

Table 1: Top-1, Top-5, and Top-1-norm accuracy of the skeleton-based action recognition for 60 and 120 classes.

action	methods	Top-1(%)	Top-5(%)	Top-1-norm(%)
60	2sAGCN(CE)[16]	40.27	72.65	24.01
	2sAGCN-C(ours)	38.42	70.29	32.89
	CTRGCN(CE)	41.47	73.84	27.32
	CTRGCN-T(ours)	38.14	65.29	31.20
	CTRGCN-C(ours)	42.94	73.00	34.20
120	2sAGCN(CE)[16]	38.41	70.49	17.56
	2sAGCN-C(ours)	37.48	69.40	28.94
	CTRGCN(CE)	39.28	70.73	20.09
	CTRGCN-T(ours)	37.21	64.81	28.05
	CTRGCN-C(ours)	40.96	72.10	30.78

The cross-entropy method and our method showed higher values in Top-1 and Top-1-norm. In particular, the difference between them can be found in the Top-1-norm. For example, CTRGCN-C(ours) is 10 percentage points better than the cross-entropy method CTRGCN(CE) for the 120-class Top-1-norm. The reason for the difference in Top-1-norm accuracy is considered that the BABEL dataset is imbalanced data. Figure 3a shows the number of data for each category. As can be seen from Fig. 3a, the number of data among the categories is biased, with the highest number of categories having 53 times more data than the lowest number of categories. Figure 3b and 3c show the confusion matrix of the Top-1 category predicted by the model and the grand truth category. In the case of the cross-entropy method, the accuracy of Top-1 tends to decrease as the number of data decreases. On the other hand, our method shows higher accuracies for the categories with a small amount of data than the cross-entropy method.

To confirm whether the accuracy varies with the degree of data imbalance, we examined the Top-1, Top-5, and Top-1-norm metrics for different numbers of categories. Figure 4 shows the Top-1, Top-5, and Top-1-norm values when the number of categories is increased in descending order by the number of data. When the number of categories is 30 and the bias in the number of data is small, the cross-entropy method is superior for the Top-1 and Top-5. However, as the number of categories increased and the bias of the data increased, our method was superior to the cross-entropy baseline in all metrics. In particular, it significantly outperformed baselines in the Top-1-norm metrics. This result indicates that contrastive learning alleviates the problem of imbalance in the number of data across categories.

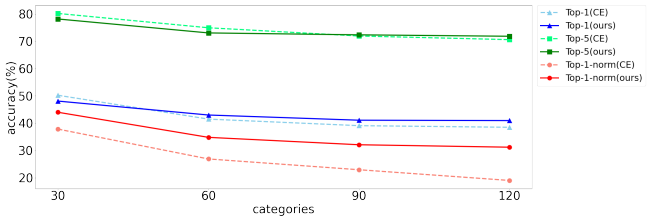
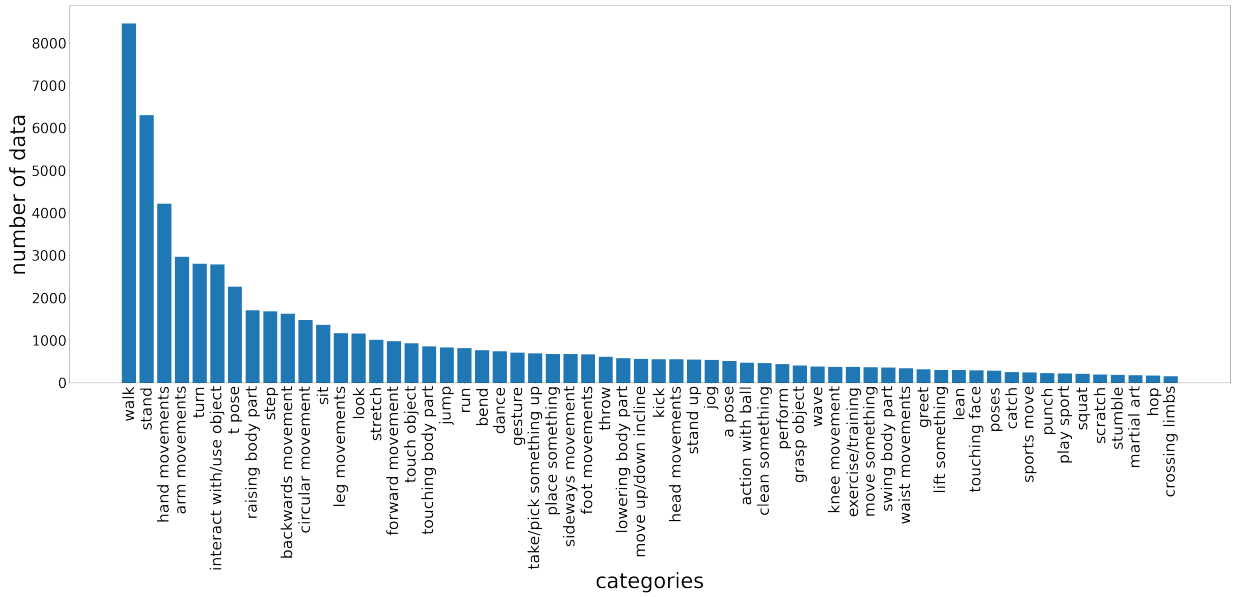


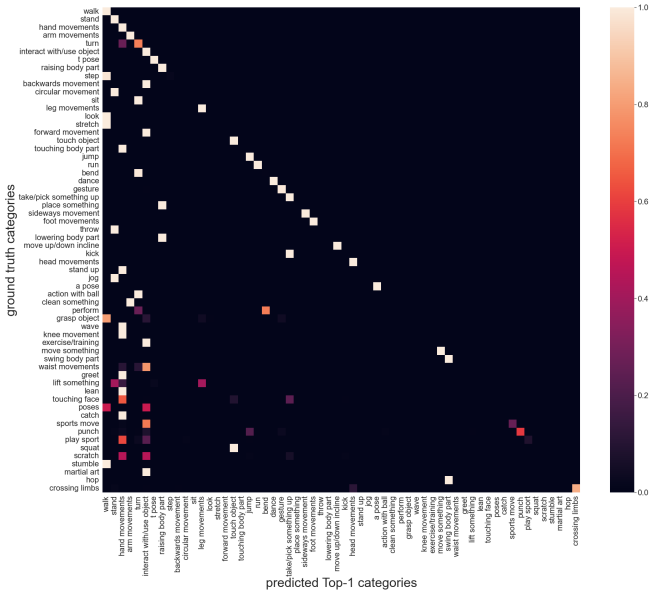
Fig. 4: Dependence of recognition performance on the number of categories. Top-1, Top-5, and Top-1-norm values are shown. The results are shown by dashed lines for the cross-entropy method and solid lines for our method.

3.3 Motion Retrieval

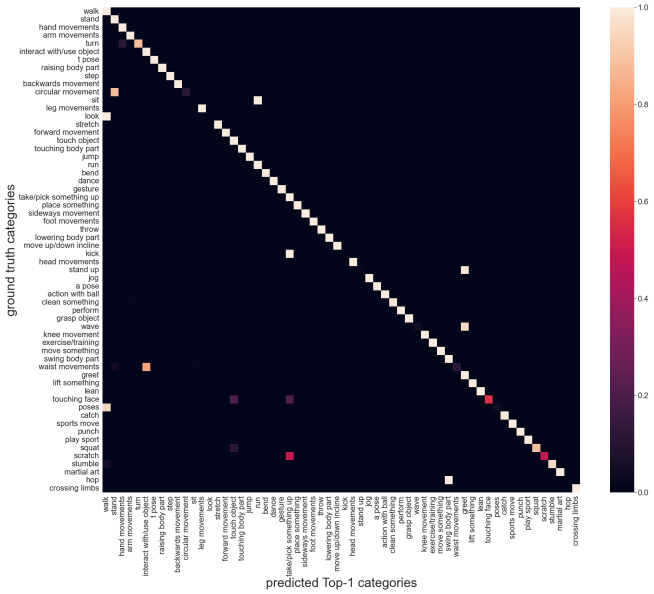
In this section, motion retrieval experiments were conducted to verify the model's ability to understand text representations of motion. In this motion retrieval, the model learns more detailed correspondence between the motion and text than in the previous experiment by using annotations instead of action categories as text. First, we trained the model using annotation as the text corresponding to the motion. After that, we performed a motion retrieval by calculating the similarity between the query text and the motion of the retrieval target using the learned model. We used arbitrarily query text, and all the validation set was used as the retrieval target. Eight query texts were used, including text specifying body parts and abstract text. The top three motions retrieved by query text and the annotations assigned to the motions are shown in Fig. 5.



(a) Number of data per category sorted in descending order

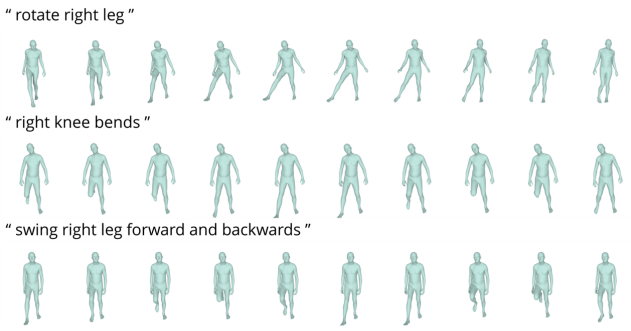


(b) CE

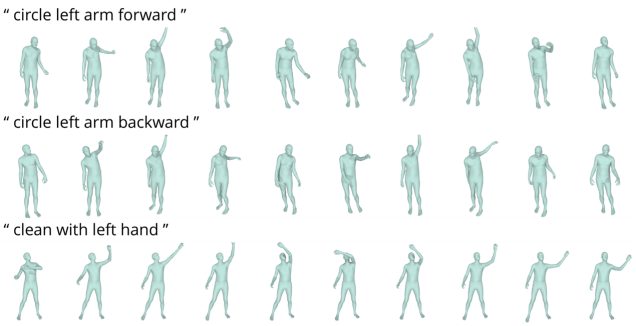


(c) ours

Fig. 3: Recognition accuracies across categories by cross entropy method (CE) and our method (ours) for imbalanced data. (b) and (c) are the Top-1 metrics confusion matrices of the model trained by cross entropy and our model trained by contrastive learning.

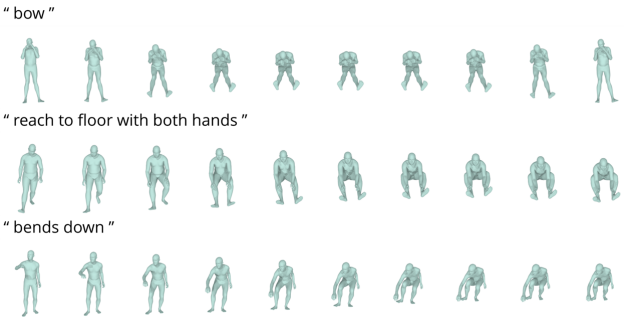


(a) move right leg

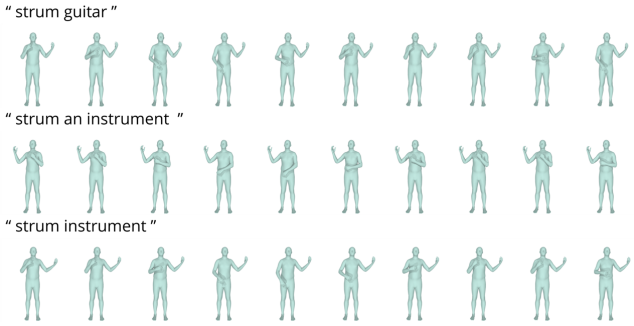


(b) move left arm

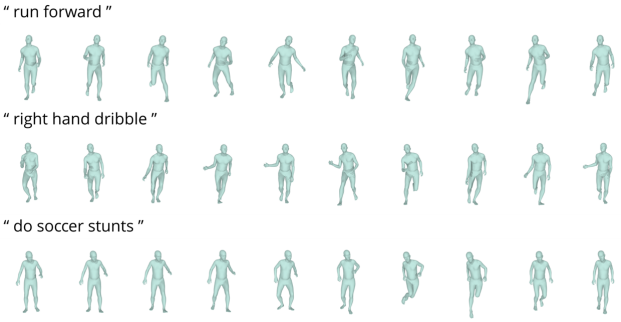
(c) high jump



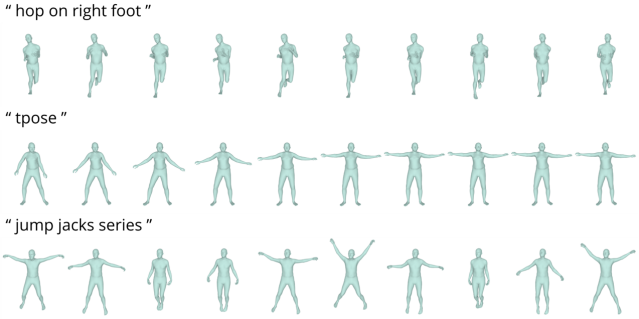
(d) bend over



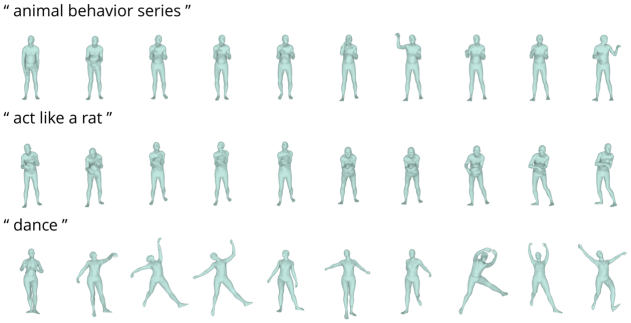
(e) play an instrument



(f) play sports



(g) repeat the same movement



(h) cute movement

Fig. 5: Visualization of the top three motions retrieved by the query text (a) to (h) with the annotations assigned to the motions.

Our model could retrieve highly relevant motions from the text that include the target body part and moving direction, such as “right leg” and “left arm” (Fig. 5a, 5b). We could also retrieve motions that involved moving the entire body from simple query text without detailed descriptions of the motion (Fig. 5c, 5d). Moreover, motions with a high degree of similarity could be retrieved from abstract query text, which does not include any descriptions to specify concrete movements (Fig. 5e, 5f). In addition, even though the query texts of “sports”, “repeat”, and “cute” had never appeared in the training data, motions that were consistent with the meaning of these queries were retrieved (Fig. 5f, 5g, 5h). This result may be because our method can use the distributed representation of the pre-trained CLIP text encoder. We also performed a motion retrieval using the model without a pre-trained CLIP text encoder (CTRGCN-T(ours)). Figure 6 shows the retrieval results by CTRGCN-T(ours). Since the model was not pre-trained to learn the correspondence between the text “sports” and the motion, motions with low relevance were retrieved. The results indicate that our model incorporates the understanding in both concrete and abstract levels of text representation for motion. The text understanding by the pre-trained text encoder was associated with motions through contrastive learning and reasonable motion retrievals were achieved. Furthermore, by applying the pre-trained distributed representation to the motion, it was possible to associate even untrained motion-text pairs.

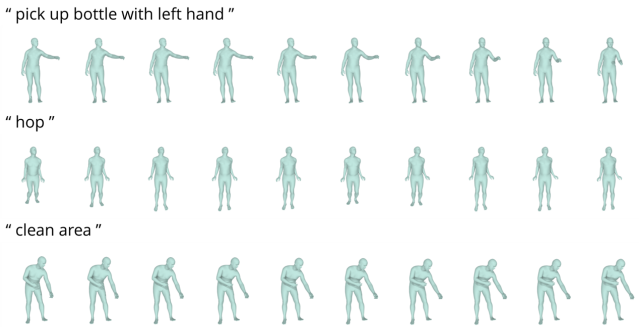


Fig. 6: Result of motion retrieval by query text “play sports” in the model using a transformer that has not been pre-trained (CTRGCN-T(ours)).

Motion can also be retrieved by measuring the similarity between the query text and the annotation assigned to the motion. We also performed motion retrieval by the query text and annotation to investigate the difference between retrievals by motion and by annotation. Table 2 shows the annotations assigned to the top three motions retrieved by motion and retrieved by annotation. The annotation similarity rank between query text and annotations of the retrieval motions is also shown. The retrieval results by motion and text have relatively lower similarity ranks than the retrieval results by annotation. This means that the motion that cannot be retrieved by the similarity of annotation can be retrieved. This was achieved by learning the correspondence between motion and text through contrastive learning.

Finally, the accuracy of the motion retrieval was evaluated using the action category. Motion retrieval was performed using the action category as query text, such as “walk” and “stand”, and it was evaluated whether the action category of the retrieved motion matched the action category as the query text (Tab. 3). Top-N is the percentage of the predictions where the ground truth category is among the top N predicted categories retrieved by the query text. We found that models with high accuracy in action recognition also had high accuracy in motion retrieval. This indicates that the accuracy of motion retrieval is correlated with the accuracy of action recognition.

4 DISCUSSION

In skeleton-based action recognition, our method significantly outperformed the cross-entropy method on the Top-1-norm metric. This is because our method can recognize even small categories of imbalanced data. The reason why the model was effective for imbalanced data is due to the use of contrastive learning. Other studies suggest that contrastive learning is an effective approach for dealing with imbalanced image data [24, 25, 26]. Our study also showed that contrastive learning is effective for imbalanced motion data while the previous studies tested the effectiveness on image data. Data-level approaches such as under-sampling are also effective for imbalanced data, but require a large amount of data. Thus, the contrastive learning used in our method has more favorable properties for motion data, with less amount of data than images due to the need for motion capture equipment for collection.

In motion retrieval, our model was able to associate concrete and abstract textual representations with motion. We also found that our model was able to infer correspondences even for untrained motion-text pairs. This is presumably because our model could obtain a high-quality embedding representation of motion by learning correspondences with a high-quality embedding representation of the pre-trained text. The use of distributed representations of pre-trained text is very effective in multimodal learning of motion and text [2, 27], and this study demonstrates its usefulness. Tevet et al. showed that by using CLIP’s text encoder, the unique embedding representations of CLIP [4], such as ”YMCA” and ”Spiderman in action!”, can be associated with motion [5]. Furthermore, by learning the correspondence between motion and text through contrastive learning, our method solves the

Table 2: The annotation assigned to the motion and annotation similarity rank retrieved by query text. The results retrieved by motion and the results retrieved by annotation are shown.

query text	retrieval method	annotation assigned to motion	annotation similarity rank
move right leg	annotation	move right leg	1
		move right leg	2
		move around move legs kick	3
	motion	rotate right leg	24
		right knee bends	369
		swing right leg forward and backwards	948
move left arm	annotation	move left arm	1
		put left arm back	2
		put left arm back	3
	motion	circle left arm forward	150
		circle left arm backward	84
		clean with left hand	520
high jump	annotation	high jump	1
		high jump	2
		high jump	3
	motion	high jump	2
		jump	32
		forward jump	57
bend over	annotation	bend over	1
		bend over	2
		bend over	3
	motion	bow	203
		reach to floor both hands	8337
		bends down	39
play an instrument	annotation	strum an instrument	1
		play guitar	2
		play the piano	3
	motion	strum guitar	1667
		strum an instrument	1
		strum instrument	421
play sports	annotation	play basketball	1
		play basketball	2
		hit	3
	motion	run forward	338
		right hand dribble	4320
		do soccer stunts	97
repeat the same movement	annotation	step back	1
		step back	2
		step back	3
	motion	hop on right foot	1719
		tpose	14529
		jump jacks series	13042
cute movement	annotation	dance move forward	1
		dance around	2
		dance around	3
	motion	animal behavior series	1445
		act like a rat	9195
		dance	93

Table 3: Top-1 to Top-100 accuracy of motion retrieval for 60 and 120 classes.

action	methods	Top-1(%)	Top-5(%)	Top-10(%)	Top-20(%)	Top-50(%)	Top-100(%)
60	2sAGCN-C(ours)	31.67	85.00	95.00	98.33	100.0	100.0
	CTRGCN-T(ours)	26.67	70.00	78.33	83.33	93.33	100.0
	CTRGCN-C(ours)	43.33	91.67	96.67	98.33	100.0	100.0
120	2sAGCN-C(ours)	20.83	65.83	75.83	79.17	85.00	90.83
	CTRGCN-T(ours)	16.67	51.67	65.00	70.83	75.00	84.17
	CTRGCN-C(ours)	25.00	70.83	80.00	83.33	91.67	93.33

problem of not understanding the moving direction of previous studies. Although we did not perform the text-to-motion generation task in this study, our model, which learned the shared embedding representation of motion and text through contrastive learning, can be transferred to the generation task.

Our model still has many limitations and potential improvements. We found that it is difficult to understand words that describe the speed of movement, such as "fast" and "slow". In addition, we fixed the motion length to five seconds in this study, so there was no validation for motions longer than five seconds with combinations of multiple movements. Besides, it is known that contrastive learning is more efficient when trained on large data sets [3, 6, 15]. Therefore, it will be necessary to increase the number of data to bring the model's accuracy to a practical level.

5 CONCLUSION

In this paper, we propose a motion-text contrastive learning method for learning shared embedding representations of motion and text. In our experiments of action recognition and motion retrieval, we found that our model learned high-quality embedding representation to make correspondence between motion and text through contrastive learning. In action recognition, our model could recognize even the action category with a small number of data in the imbalanced motion dataset. In motion retrieval, by learning the correspondence between actions and text made it possible to retrieve the motion with high similarity to concrete and abstract query text from a large amount of data. Furthermore, the model could associate untrained motion-text pairs using the pre-trained text embedding representation.

REFERENCES

[1] Chaitanya Ahuja and Louis-Philippe Morency. Language2pose: Natural language grounded pose forecasting. In *2019 International Conference on 3D Vision (3DV)*, pages 719–728. IEEE, 2019.

[2] Anindita Ghosh, Noshaba Cheema, Cennet Oguz, Christian Theobalt, and Philipp Slusallek. Synthesis of compositional animations from textual descriptions. *CoRR*, abs/2103.14675, 2021.

[3] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[4] Gabriel Goh, Nick Cammarata †, Chelsea Voss †, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. Multimodal neurons in artificial neural networks. *Distill*, 2021. https://distill.pub/2021/multimodal-neurons.

[5] Guy Tevet, Brian Gordon, Amir Hertz, Amit H Bermano, and Daniel Cohen-Or. Motionclip: Exposing human motion generation to clip space. *arXiv preprint arXiv:2203.08063*, 2022.

[6] 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.

[7] Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. A survey on contrastive self-supervised learning. *Technologies*, 9(1):2, 2020.

[8] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673, 2020.

[9] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[10] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In *European semantic web conference*, pages 593–607. Springer, 2018.

[11] Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, 2018.

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[13] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[14] Jiang Wang, Yang Song, Thomas Leung, Chuck Rosenberg, Jingbin Wang, James Philbin, Bo Chen, and Ying Wu. Learning fine-grained image similarity with deep ranking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1386–1393, 2014.

[15] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. *Advances in neural information processing systems*, 29, 2016.

[16] Abhinanda R. Punnakkal, Arjun Chandrasekaran, Nikos Athanasiou, Alejandra Quiros-Ramirez, and Michael J. Black. BABEL: Bodies, action and behavior with english labels. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 722–731, June 2021.

[17] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, October 2015.

[18] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *International Conference on Computer Vision*, pages 5442–5451, October 2019.

[19] Yuxin Chen, Ziqi Zhang, Chunfeng Yuan, Bing Li, Ying Deng, and Weiming Hu. Channel-wise topology refinement graph convolution for skeleton-based action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13359–13368, 2021.

[20] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12026–12035, 2019.

[21] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[22] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015.

[23] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016.

[24] Yassine Marrakchi, Osama Makansi, and Thomas Brox. Fighting class imbalance with contrastive learning. In Marleen de Bruijne, Philippe C. Cattin, Stéphane Cotin, Nicolas Padoy, Stefanie Speidel, Yefeng Zheng, and Caroline Essert, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2021*, pages 466–476, Cham, 2021. Springer International Publishing.

[25] Peng Wang, Kai Han, Xiu-Shen Wei, Lei Zhang, and Lei Wang. Contrastive learning based hybrid networks for long-tailed image classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 943–952, 2021.

[26] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *Eighth International Conference on Learning Representations (ICLR)*, 2020.

[27] Minori Toyoda, Kanata Suzuki, Hiroki Mori, Yoshihiko Hayashi, and Tetsuya Ogata. Embodying pre-trained word embeddings through robot actions. *IEEE Robotics and Automation Letters*, 6(2):4225–4232, 2021.