Multimedia Tools and Applications (2024) 83:70523–70544
https://doi.org/10.1007/s11042-024-18238-4

Contrastive disentanglement for self-supervised motion style transfer

Zizhao Wu[1] · **Siyuan Mao**[1] · **Cheng Zhang**[1] · **Yigang Wang**[1] · **Ming Zeng**[2]

Received: 15 June 2023 / Revised: 2 January 2024 / Accepted: 7 January 2024 /
Published online: 30 January 2024

Abstract
Motion style transfer, which aims to transfer the style from a source motion to the target while keeping its content, has recently gained considerable attention. Some existing works have shown promising results but required labeled data for supervised training, limiting their applicability. In this paper, we present a novel self-supervised learning method for motion style transfer. Specifically, we cast the problem into a contrastive learning framework, which disentangles the human motion representation into a content code and a style code, and the result can be generated by compositing the style code of source motion and the content code of target motion. To encourage better code disentanglement and composition, we investigate InfoNCE loss and Triplet loss in a self-supervised manner. This framework aims at generating reasonable motions while guaranteeing the disentanglement of the latent codes. Comprehensive experiments have been conducted over the benchmark datasets and demonstrated our superior performance over state-of-the-art methods.

Keywords Motion style transfer · Human motion analysis · Disentangled representation learning · Neural networks

✉ Zizhao Wu
wuzizhao@hdu.edu.cn

Siyuan Mao
siyuanmao@hdu.edu.cn

Cheng Zhang
zhangcheng828@foxmail.com

Yigang Wang
yigang.wang@hdu.edu.cn

Ming Zeng
zengming@xmu.edu.cn

[1] School of Digital Media Technology, Hangzhou Dianzi University, Hangzhou 310018, China

[2] School of Informatics, Xiamen University, Xiamen 361005, China

1 Introduction

Nowadays, multimedia data is ubiquitous in daily life, ranging from video games, robotics, virtual reality (VR), animated films, and many others. Among these various forms of multimedia data, the human-like characters and human motion are core aspects of these data and applications. However, the traditional pipeline of human motion creation is notorious for its time-consuming and tedious manual tweaking. Thus the rapid generation and flexible reuse of an existing motion are of significant importance to enrich human motion production.

Generally, human motion is considered to be composed of two factors, termed content and style [1]. Content usually represents the nature of a motion, while style represents spatio-temporal variations of a movement that relates to identity, personality, and emotion. Figure 1 shows an example, where both input motions are walking in content, but the upper carries *depressed* style, while the lower equips with the *angry* style. This creates an open issue, which is to transfer the style from an existing motion to another while preserving the content of the latter, i.e., motion style transfer.

Recently, witnessing the significant success of deep learning in computer vision tasks, many researchers have explored deep neural networks for style transfer tasks, i.e. deep motion style transfer. Among these works, most of them are purely data-driven [2, 3], i.e., leveraging a large collection of training data to learn the latent style representation for style transfer. Though effective, these methods have encountered problems like data scarcity and data bias. To alleviate these issues, Aberman et al. [4] have recently suggested a supervised motion style transfer method from unpaired data, i.e. video to animation. In their approach, a neural network is designed to disentangle the content and style from the unpaired motion sequences in favor of style transfer. Though effective, this method depends on the style labels to disentangle the style features. Then Pan et al. [5] present an unsupervised motion style transfer method based on a meta network, which takes a style motion as input and generates natural stylized motion via propagation on the transform network with a content motion. In their approach, the style is simply characterized as the mean-variance statistics between the transferred motion and style motion, which limits their applicability for generalization [6], especially in processing with unseen styles and sequence data, due to the entanglement

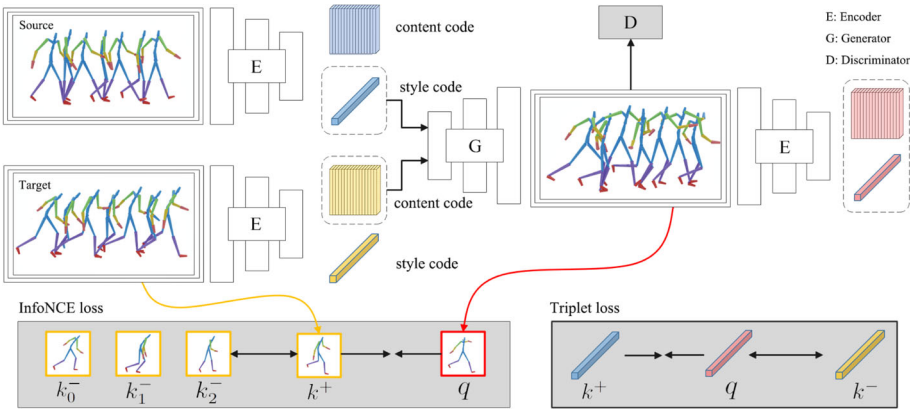


Fig. 1 The architecture of our motion style transfer system, wherein the black arrows pointing to each other pull the samples closer, the black arrows pointing away push them apart, the yellow and red arrows reveal sampling strategies. We also use the square boxes to denote the content codes, and employ the strips to denote the style codes respectively

between their content code and style code. More recently, Park et al. [7] and Jang et al. [8] present methods based on spatio-temporal graph convolution networks (ST-GCN) [9] to capture features, which can transfer arbitrary style motions without labels, i.e. unseen styles. Though effective, these methods contain several pooling and unpooling processing to extract skeleton information at body part level may lead to the discarding of key joint data.

In this paper, we propose a novel motion style transfer method that learns to disentangle style and content of arbitrary motions in a self-supervised manner. Our work is inspired by the recent work [10, 11] on content and style disentanglement for image style transfer. According to our knowledge, no prior work can reach the disentangled representation in an unsupervised manner for the motion style transfer task. Constructing disentangled representations is known to be a difficult task, especially in the unsupervised scenario, achieving such representations would enable us to perform complex and highly useful operations on the data [12]. In our task, learning to disentangle style and content of arbitrary motions will enable us to perform more flexible manipulation of styles, like style transfer, style interpolation, etc. We achieve this goal through contrastive learning, a powerful tool that learns representation functions to map semantically similar data closer in the latent space.

Specifically, given a pair of motions, our network first encodes them into content and style codes which are then respectively rearranged to feed into a generator, for the creation of new motion carrying transferred style. Two types of the contrastive loss function have been investigated, namely InfoNCE loss and Triplet loss, which aim to ensure the disentanglement of the content code and the style code in the latent space, respectively. To achieve that, both the InfoNCE loss and Triplet loss sample positives and negatives from the codes of input motions and generated motions, which can be thought as sampling internally, rather than sampling externally from other labeled motions. In this way, our approach is self-supervised to bypass the requirement for the labeled data. Experiments have been conducted on three commonly used datasets, and the results verify the effectiveness of our framework by comparing it with some state-of-the-art approaches.

In summary, the main contributions of this paper are as follows:

1. We present a novel self-supervised learning approach for motion style transfer. According to our knowledge, we are the first to learn disentangled representation in an unsupervised manner in the paradigm;
2. Our model learns a disentangled representation through a novel contrastive learning framework;
3. Our method achieves superior performance comparing with several SOTA models.

2 Related work

2.1 Human motion synthesis

The human motion synthesis is closely related to our topic, as both of them need to generate new motions while keeping some constraints. In the early days, supported by the existence of large human motion databases, many database-driven approaches have been investigated, including motion graph [13–15], independent component analysis [16], Gausssian process based models [17–20], Bayesian network [21], eigen analysis [22] and so on [23–25].

Recently, with the surge of deep learning, a variety of neural networks have been used to synthesize human motions. For example, Zhou et al. [26] investigated an auto-conditioned recurrent neural network to synthesize complex human motion. Martinez et al. [27] employed

recursive neural networks to facilitate the short-term motion prediction and long-term human motion synthesis. Jain et al. [28] investigated to combine high-level spatio-temporal graphs with RNNs for multiple tasks including human activity detection and prediction. Variations of GANs [29, 30] and adversarial training [31, 32] have also been applied for motion synthesis.

2.2 Motion style transfer

To better characterize motion, many researchers have exploited the high-level abstract concept, i.e. style, to enrich human motion generation. Rose et al. [33] have studied emotion as a driver of motion style. Hoyet et al. [34] investigated what factors that make human motion recognizable and appealing, and figured out with the concept of distinctiveness and attractiveness to evaluate the human motion. Kiiski et al. [35] examined that visual information of human body motion has relation with social traits and intention perception, and also the human motion features can be used for social intention prediction. Akin to these works, many researchers [36–38] have also investigated the connection between motion style and personality perceptions. To conduct motion style transfer, i.e., transferring the motion style from one motion sequence to another, while holding the motion content of the latter, many researchers have exploited many tools, like linear time-invariant model [39] and Gaussian process model [40], to proceed with this problem.

Recently, witnessing the significant performance of convolutional neural networks on computer vision tasks, and also the development of image style transfer technology [41], Holden et al. [42] pioneered a deep auto-encoder framework by taking the control signal as inputs, and performed motion style transfer via image-based style transfer technique [43]. In their follow-up work [2], they have suggested a novel framework, which consists of two structures: a loss network to train the convolutional autoencoder, and a feed-forward network to perform motion style transformation. However, this work simply employed the gram matrix to define the style loss, which leads to a strong dependency on both content and style.

To overcome this issue, diverse neural network architectures [32, 44, 45] have been exploited. For example, Smith et al. [44] suggested a fast style transfer network which consists of three controllable network modules to learn from data and produce stylized motion. They also provided an auxiliary one-hot style, which enables the user to easily control the designed style interpolating and blending. Most notably, Aberman et al. [4] provided an unpaired motion style transfer approach from video to motion. In their approach, an asymmetric network is devised with several loss functions to guide the generation of stylized motions. Recently, Tao et al. [46] presented an effective transformation model which can be applied to operate both online and offline. And they introduced the framework of Encoder-Recurrent-Decoder to handle this task for strengthening the connection and relationship of temporal succession information. However, one big obstacle to the above-mentioned approaches is that they are supervised learning-based, i.e. labeling data is required in their approaches, which hinders their applicability. Before long, Pan et al. [5] presented an unsupervised motion style transfer approach that employs a meta network to generate another transformation network. However, In their approach, the motion style feature is computed based on the mean-variance statistics between the transferred motion and style motion, which will lead to severe flickering artifacts for unseen style and sequence data [6], and what's more, the content and style features are entangled. Moreover, Wen et al. [47] attempted to utilize a generative flow model to process content codes and style codes from input motion, without any separate preprocess, causing the same problem as above. As a comparison, our method explicitly factors out the latent style codes based on a style encoder, which benefits the manipulation of styles,

including style transfer. Very recently, inspired by the ST-GCN [9], Park et al. [7] and Jang et al. [8] proposed an unsupervised network that considers the spatial relationship between joints. But the integration of this framework also introduces a new problem, i.e. that pooling and unpooling layers used in this framework will cause the loss of certain information of the key joints. Comparing with the aforementioned methods, our method proposes a novel contrastive learning framework to generate motions with high fidelity and quality.

2.3 Disentangled representation learning

Recently, some authors have argued that disentangled representations are appealing in learning to solve challenging real-world down-stream tasks [12, 48, 49]. Afterwards, many approaches have been proposed to force the emergence of disentanglement in the learned representations. For example, Gatys et al. [43] pioneered to learn an image representation to separate content and style. Liu et al. [50] trained an auto-encoder model supervised by ground truth labels to learn a content-invariant representation. Recently, learning disentangled representations from unlabeled data has gained increasing popularity. Among these works, Variational AutoEncoder (VAE) and its variants [51–53] have dominated the recent advances in the filed.

Disentangling sequence data into time-varying and time-independent components has also been explored in many settings. For example, some authors [54, 55] have proposed an auto-encoder architecture for next frame prediction of videos, with two separate encoders responsible for content and motion at each frame.

In this work, we introduce a novel disentangled representation learning approach for motion style transfer.

2.4 Contrastive learning

Contrastive learning has recently been widely used to learn rich representations of given input data. The core idea is to learn representation function that maps semantically similar data closer in the embedding space. For instance, Contrastive Predictive Coding (CPC) [56] pioneered the practice of learning long-term relations and predicted the latent representation of the future part. Momentum Contrast (MoCo) [57] built dynamic dictionaries for contrastive learning and leveraged the instance discrimination for unsupervised image feature learning.

To date, the choice of negatives is one of the key challenges in contrastive learning paradigm, which determines the quality of the underlying representations learned and the loss functions used. Based on different strategies, there are many classic methods, such as InfoNCE [56], Triplet [58], Siamese [59] and so forth.

3 Method

The primary goal of our method is to transfer the style from a source motion to another, while preserving the content of the latter. Given the source motion clip $M^S \in R^{n\times d}$ and target motion clip $M^T \in R^{n\times d}$, our objective is to generate the re-target motion clip $M^G \in R^{n\times d}$ that carries the content of M^T, while performing with the style of M^S. Here, n is the temporal dimension size, and d refers to the spatial dimension size for each frame.

To reach the above goal, we propose a novel contrastive disentanglement framework, which contains the encoder and the generator blocks, as is illustrated in Fig. 1. Specifically,

each encoder in our framework consists of the content encoder E_c and the style encoder E_s, thus the latent code of each motion is factorized into a content code and a style code. Style transfer is finally achieved by recombining the style code of the source motion to the content code of the target motion in the generator block. To ensure the learned representation carries useful information of original data as much as possible and ensure the disentangled representation's reasonableness, our method investigates a novel self-supervised framework containing the reconstruction learning and contrastive learning modules. Below we describe the details.

3.1 Reconstruction learning

Without loss of generality, given the original input data $x \sim M$, the latent representation $z \sim Z$ can be learned through an encoder $E(x)$. In order to make sure that the learned representation carries valuable information of input data as much as possible, z is required to be able to reconstruct the input through a decoder $G(z)$. The reconsturcted data can thus be defined as: $\tilde{x} = G(E(x))$, which is the basic model of auto-encoder in representation learning.

We follow the classic autoencoder framework [2, 5, 42] to learn the latent representations of given motions. However, as mentioned before, we intend to disentangle representation into two latent codes, content code and style code. Thus different from classic autoencoder, we design two encoders that encode the original motion into two latent codes $E_c(x)$ and $E_s(x)$, and utilize the combination of $[E_c(x)|E_s(x)]$ as a joint latent representation to reconstruct the input motion. The reconstruction loss can therefore be defined as the mean Euclidean distance between the input motion and the reconstructed motion based on the encoders E_c and E_s, and the decoder G:

$$\begin{aligned}\mathcal{L}_{rec}(E_c, E_s, G) = \mathbb{E}_{M^S \sim X}[\|G(E_c(M^S)|E_s(M^S)) - M^S\|] \\ + \mathbb{E}_{M^T \sim X}[\|G(E_c(M^T)|E_s(M^T)) - M^T\|],\end{aligned} \tag{1}$$

where | is an operator that concatenates $E_c(x)$ and $E_s(x)$ in our setting. To reduce the reconstruction loss, the encoders E_c and E_s tend to encode as much information as possible, while the decoder G encourages that the latent code $E_s(x)$ contains sufficient information to represent the original data.

Moreover, we employ a discriminator module [10, 29] to help the decoder to reconstruct more realistic output. Conceptually, a typical discriminator model comprises two models pitted against one another: a generative model, G, is trained to generate plausible data, i.e. motions resembling the real data distribution, and a discriminative model, D, is trained to distinguish the generator's fake data from real data. The discriminator penalizes the generator for producing implausible results. In our case, we utilize the decoder G of the above autoencoder as the generator and introduce a Multi-Layer Perceptron (MLP) as the discriminator, which tries to distinguish real data from the data created by the generator. These two models, generator and discriminator, can be seen as oppositions as their objectives are in competition with one another, and form a Generative Adversarial Network (GAN). The adversarial loss is represented as:

$$\begin{aligned}\mathcal{L}_{GAN}(E_c, E_s, G, D) = \mathbb{E}_{M^S, M^T \sim X}[1 - \log(D(M^G))] \\ + \mathbb{E}_{M^S \sim X}\| \log(D(M^S))\|,\end{aligned} \tag{2}$$

where $D[\cdot]$ refers to the discriminator function, and $M^G = G(E_c(M^T)|E_s(M^S))$ represents the generated motions. This loss function encourages the generated motion to be realistic.

3.2 Contrastive learning

Despite that the above constraints yield a factored representation, the resulting representation can not be guaranteed that E_c and E_s actually encode the content and style. To resolve this, inspired by the recent work [10, 60] on contrastive learning for disentanged representation learning, we investigate contrastive learning to ensure the interpretability of the disentangled representations in a self-supervised manner, which is the key to our framework. Toward the objective definition, we enforce that the latent codes of $E_c(M^G)$ and $E_c(M^T)$ are similar, and so are the latent codes of $E_s(M^G)$ and $E_s(M^S)$. In our implementation, we define the latent content code $E_c(\cdot)$ to be a tensor with size of $h \times t$, while the latent style code $E_s(\cdot)$ is defined to be a vector with the size of l, in light of that style is irrelevant to content.

Typical workflow of contrastive learning is to pull positive pairs together while pushing negative pairs apart, where *query* and its *positive* form the positive pair, and *query* and its *negative* form the negative pair. In our context, we refer *query* to an output, *positive* and *negatives* are corresponding and noncorresponding input.

Content code To achieve the disentangled content code, we note that the generated motion should be similar to the target motion in terms of content. Therefore, it is straightforward for us to define the positive pair, which includes the sampled content code of the generated motion as the *query* and the corresponding part of the content code of the target motion as the *positive*. On the other hand, how to define the *negatives* is crucial for contrastive learning. In this work, we observe that within one motion data, the content of motion clips is diverse in general, which inspires us to draw negative pairs internally from within the target motion. Specifically, we sample content code of other irrelevant parts of the target motion as the *negatives*. See the Fig. 1 for an illustration.

Formally, we use $q, k^+, \{k_j^-\}_{j=1}^n \in R^h$ to denote the query, positive and multiple negative samples from $E_c(M^G)$ and $E_c(M^T)$. Let $i(1 \leq i \leq t)$ be one randomly sampled value, q represents the sampled i-th vector of $E_c(M^G)$. Accordingly, k^+ indicates i-th vector of $E_c(M^T)$, we then sample $\{k_j^-\}_{j=1}^n$ from the other components of $E_c(M^T)$ except i, i.e. k_j^- denotes j-th ($j \neq i$) vector of $E_c(M^T)$. We investigate the InfoNCE loss to encourage that q matches with k^+ comparatively more than any of the keys k^-. By measuring the query-key similarity, we reach the InfoNCE loss in a softmax formulation:

$$\mathcal{L}_{nce}(E_c, E_s, G) = -\mathbb{E}_{q,K\sim E_c(X)}[\log \frac{exp(f(q,k^+))}{\sum_{j=1}^n exp(f(q,k_j^-))}], \tag{3}$$

where function $f(q,k)$ measures the similarity score between the query q and the key vector k, and is usually modelled as bilinear products, i.e. $q^T Wk$, where W is a linear predicted transformation [61].

Style code To obtain the disentangled style code, akin to former strategies [4, 10, 60] we can set up the positive pair, which includes the sampled style code of the generated motion as the *query* and the corresponding part of the style code of the source motion as the *positive*. However unlike InfoNCE loss, it is hard to sample negatives in a former similar way as style code is abstractly defined without ordering.

We alternatively resort to Triplet loss which uses one negative pair instead of negative pairs of InfoNCE, with corresponding to one positive pair for modeling, in order to make sure that the sampled positive pair are closer than that of negative pair in the latent space.

Concretely, assume we have an encoded query q indicating the style code $E_s(M^G)$ from the generated motion M^G, and a pair of encoded key vectors $K = (k^+, k^-)$ consisting of a positive sample $k^+ \in R^l$ representing the style code $E_s(M^S)$ of the source motion M^S, and a negative sample $k^- \in R^l$ representing the style code $E_s(M^T)$ of the target motion M^T. Triplet loss aims to pull positive samples into nearby points while pushing negative samples apart from each other, in the latent space. We also draw an illustration in Fig. 1. The optimization target of the Triplet loss is:

$$\begin{aligned}\mathcal{L}_{tri}(E_c, E_s, G) =& \mathbb{E}_{q,k^+,k^- \sim E_s(X)}[\|E_s(q) - E_s(k^+)\| \\ & - \|E_s(q) - E_s(k^-)\| + \lambda]_+,\end{aligned} \tag{4}$$

where $[\cdot]_+ = max(0, \cdot)$ represents the hinge loss function, and λ is the violate margin that enforces the distance of positive pairs to be closer than the negative pairs, in our setting, we set the value to 1.0.

Overall loss function The overall loss function can be calculated as the weighted sum of all the terms described above.

$$\mathcal{L}_{total} = \mathcal{L}_{rec} + \alpha\mathcal{L}_{gan} + \beta\mathcal{L}_{nce} + \gamma\mathcal{L}_{tri}, \tag{5}$$

where α, β and γ are the weighting factors for the loss functions of adversarial, infoNCE and Triplet, individually. In our experiments, we use $\alpha = 1$, $\beta = 0.3$, and $\gamma = 0.3$.

In this section, we first present the datasets used in our experiments. Then, we validate the reasonableness of the disentangled codes through visualization. Finally, some visual results are provided, as well as the comparison with some state-of-the-art methods to demonstrate the capacity of method.

4 Experiments and evaluation

4.1 Dataset and implementation

Dataset We consider two widely used motion capture datasets: The first dataset is CMU Mocap dataset (CMU) [62], which contains 2605 unique motion sequences captured by an optical motion capture system. These sequences are roughly categorized into 144 subjects and we select sequences of motion with specific style such as *elated*, *march*, *injured*. The other dataset is XIA [63], which contains motion clips with 8 style labels.

In our implementation, we trim the motion sequences from each of the dataset into shot overlapping clips with T=32 frames and an overlap of $T/4$, which leaves 12005 motion sequences for the CMU dataset, 1500 motion sequences for the XIA dataset. These motion clips are further organized into two disjoint training and testing sets, where the testing set accounts for 10% of the total number, and the rest are training set.

Data pre-processing and post-processing In our work, following the work of Aberman et al. [4], we first scale all the data to uniform space, then, for each data sample, we concatenate joint positions and joint rotation as human pose representation of our model. Precisely, the joint positions are represented by the local coordinates, and the joint rotation is represented

by unit quaternions. In our implementation, a motion clip $m \in \mathbb{R}^{n \times d}$ consists of n poses, each pose is represented by $d = 7J$ channels, where $4J$ is used for joint rotations and $3J$ is used for joint positions, J denotes the number of joints.

In our model, we predict the joint positions and joint rotations straightforwardly from the network. Therefore it is hard to guarantee the joint lengths of the skeletons. In addition, the subtle noise on the foot joint will make the whole pose slip on the plane. To alleviate these issues, we post-process the motion clips by extracting the foot contact clips from the target motion and correct their feet positions and further apply the Inverse kinematics (IK) to correct the bone length of the generated motion clips. We note that this step is common in other baseline approaches [4, 5].

Implementation details The encoders E_c and E_s of our model consist of 3 layers of 1D convolution, followed by 3 residual blocks. The generator G consists of 4 residual blocks with AdaIN. The content code in our network is a tensor with temporal dimensions of 144×16; the style code is a 144-dimensional vector. In addition, we set the batch size to 16, and optimize the algorithm with a learning rate of 0.0001. Finally, our method is implemented in PyTorch and requires 200000 iterations to coverage on an NVIDIA GTX2070 GPU.

Metrics We perform a variety of experiments to estimate the competence and effectivity of our model by utilizing three kinds of metrics: Style Distance, Content Distance and Fréchet Motion Distance (FMD), Kernel Inception Distance (KID), and Diversity.

In line to the measurements of Gatys et al. [43], the content is described by the features embedded in neural layers of the network, and the style is represented by the Gram matrix over these features. Following their jobs, after training on CMU dataset, we compute the Gram matrix difference between the generated motion and the original style motion, and between the generated motion and the original content motion, aiming to measure the style distance and the content distance separately.

In addition, FMD is firstly proposed in [7], inspired by a plausible metric in the evaluation of image generative field, namely Fréchet Inception Distance (FID) [64], which is available to measure the distance between the original motion and the generated motion. Hence FMD can calculate the concrete difference between the feature distributions of the fake motion and the real motion, which is extracted by the feature extractor. As no standard feature extractor of motion, we train a denoising autoencoder to extract the latent information for the calculation of FMD, following [46]. The lower FMD, the better performance on generative capability and transformation expression of the model.

The Kernel Inception Distance (KID), introduced by Binkowski et al. [65], assesses skewness and compares mean and variance similar to FID. KID demonstrates superior performance with small and medium-sized datasets, where lower values indicate better performance.

The Diversity metric evaluates the variability of generated movements. Specifically within action recognition models, it examines variance across all action categories, making it suitable for an unrestricted generator. A high diversity value, approaching that of the ground truth, is considered favorable.

4.2 Visualization

In order to achieve an intuitive concept of how the content and style depicted in the network, in this subsection, we try to project the disentangled codes onto a 2D embedding space by adopting T-distributed Stochastic Neighbor Embedding (t-SNE) algorithm.

Style embedding Figure 2(a) shows the style embedding of our network, in which each dot represents a motion. In order to evaluate the quality of the network, we collect the style labels defined on the dataset, and marked each point with a color corresponding to its style label. We can see that the coloring distribution is consistent with the clusters, which means the network encourages that the style features belonging to the same style label have similar representations. In Fig. 2(b), we illustrate the style embedding without the contrastive loss in our loss functions, and the embedding results without $\mathcal{L}_{nce}$ or $\mathcal{L}_{tri}$ are demonstrated in Fig. 2(c) and (d) below. We can see that the boundaries of each cluster are not clear and even intersect with each other. This strengthens usefulness of the contrastive loss.

Unseen style embedding To evaluate the potential of our model to unseen style, we exclude motions that are labeled by depressed style label from the XIA dataset, and retrained the model. Then we test the unseen motion in our experiments to valid the generalization performance of our model, and also make a comparison with Pan et al. [5]. Figure 3 shows the unseen style embedding. We can see that our model in (b) successfully adapts the unseen style to other motions, whereas the method of [5] in (a) can hardly identify different styles. In Fig. 4, we present the generated motions from unseen styles, the results are natural and desirable.

4.3 Evaluation results

We conduct experiments on the aforementioned datasets, and compare the results with some state-of-the-art methods in this subsection.

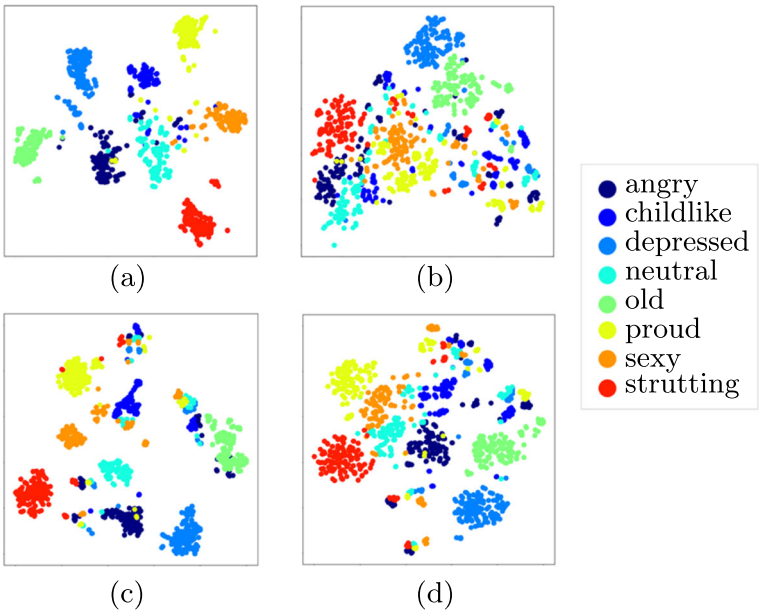


Fig. 2 Effect of contrast loss in style. The result is trained with contrast loss in style (a) and without contrast loss in style (b). Moreover, the visualization comparison of model capability without $\mathcal{L}_{nce}$ or $\mathcal{L}_{tri}$ has been shown below in (c) and (d), respectively

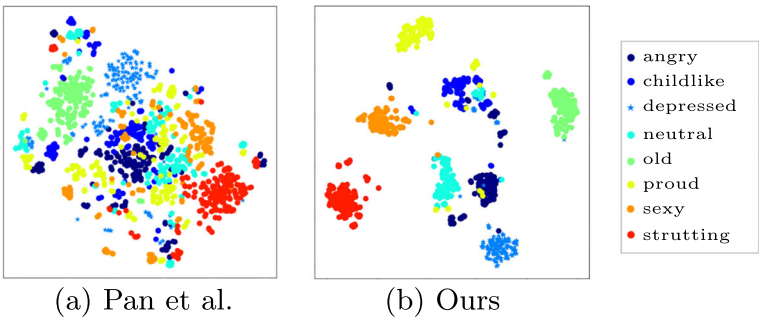


Fig. 3 Unseen style embedding. Trained on XIA dataset excluding the *depressed* style. Compared with the method of [5], as is shown in (a), our method in (b) can better distinguish different styles

Evaluation Figure 5 shows the visual results taken from the output of the network. The top row presents the source motions, which contain a variety of different styles, such as *depressed* and *old*. The second row lists the target motions which contain a variety of content. In the bottom row, we show the results after conducting motion style transfer. From the bottom row, we can see the results are favorable in carrying the styles from the top row and the content from the middle row. The results validate the effectiveness of our method to transfer one style from a motion sequence to another.

In Fig. 6, we make a qualitative comparison with the state-of-the-art methods. The leftmost column provides four clips from different content motions. The second column gives four clips from different style motions. The next five columns showcase the transferred results of the five methods [4, 5, 7, 42, 46] for comparison, and the rightmost column depicts our generated motion clips. In the results, we colored the arm bones to highlight the differences among the results.

From the visual results, we can observe that the method of Holden et al. [42] struggles to produce plausible motions. We can notice that the content of generated motion varied with the original content motion. Besides, the generated motion clips are also not consistent with ergonomic, as some irrational joints are highlighted with black circles in the third column. This is because that their method learns a shared representation for both the content and the

Fig. 4 Unseen style transfer. The output motion clips generated from the proposed network with unseen *depressed* style. As can be seen from the figure, albeit the contents are different, in both cases the motions are plausible and the target style has been identified

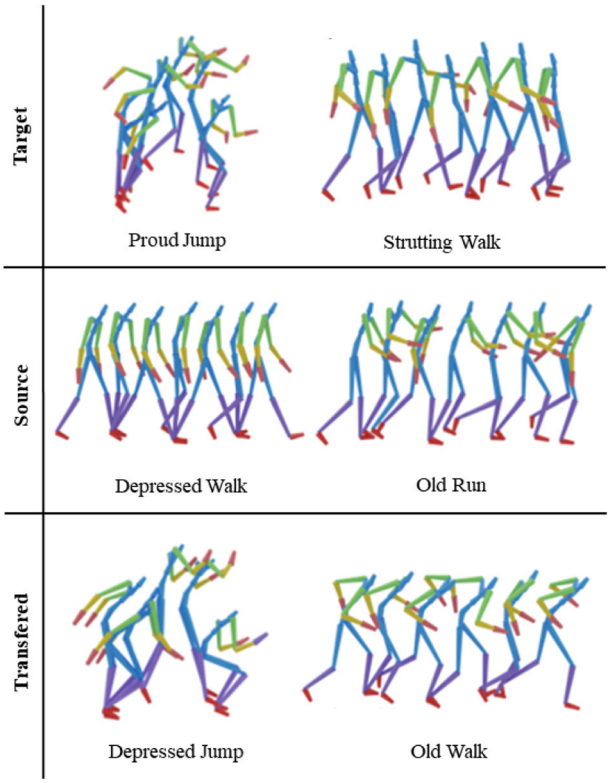


Fig. 5 Samples of our style transfer results. The styles of the source motion (top row) is transferred to a motion with target content (middle row) and generates new motion (bottom row), carrying the desired content and style. We marked its content and style in the lower of each motion

style. When performing style transfer, such representation will bring mixed content and style information to the target motion.

Comparing with the approach of Aberman et al. [4], though both approaches visually preserved the content of the source motions, our approach generates more plausible results. Specifically, in the fourth column, we highlight the distinctive joints with black circles. As can be seen from these joints, the joints generated by Aberman et al. [4] exist large joint angles, which are hardly seen in natural human motion sequences. Our method is instead more plausible. Moreover, we emphasize that this approach requires style labels, whereas our method does not need.

Comparing with the approach of Pan et al. [5] in the fifth column, we note that there exists inconsistency in the generated motions, i.e., adjacent motion clips may exhibit large variations in pose and style. We argue that their model simply discards contrastive learning module when comparing with Aberman et al. [4], which leads to poor discrimination ability for style. Instead, our method has made great progress in this aspect.

The visual representations of Park et al. [7] are shown in the sixth column, there are partial results which contain certain unnatural body postures and unreasonable joint angles,

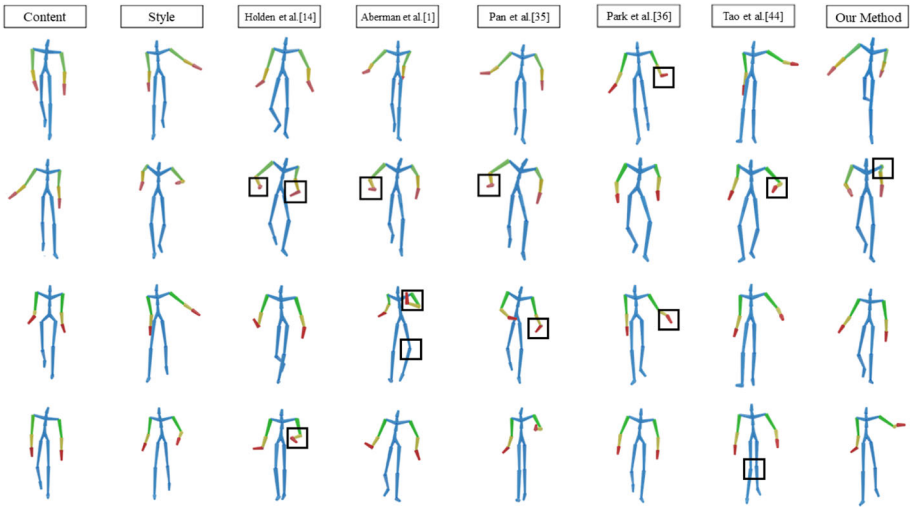


Fig. 6 Visual comparisons of our method and previous methods. The leftmost column lists motion clips in which we need to borrow their content. The second column illustrates styles that we need to transfer. The last six columns show the generated motion clips based on different algorithms

due to the information missing existed in ST-GCN [9]. Comparing with the results of Tao et al. [46], we notice that the results exist the same problems as above, i.e. distorted joints, which are marked by the black circles in the seventh column. The reason behind that is their approach pays attention to improve the speed and efficiency of model, while ignoring the quality and exactitude of generated results as shown in the figure. And their approach just offers a supervision pattern for this task, however we don't need labels for help but transfer motion with high fidelity.

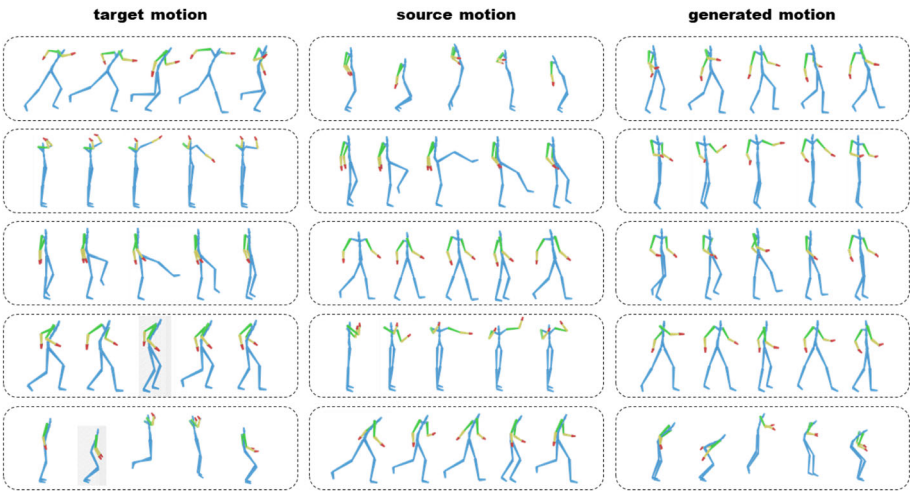


Fig. 7 More visualization results of our method on Xia dataset

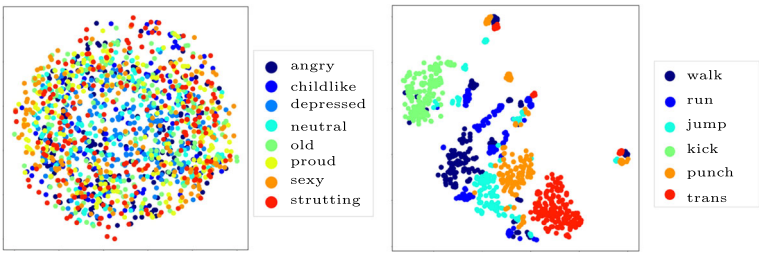


Fig. 8 We project the content codes of our test samples onto 2D space using t-SNE. (a) The samples are colored based on the style labels which take motions that cover multiple categories of content and multiple categories of styles as input. (b) The samples are colored based on the content labels which take motions that covering with multiple categories of content and identity style as input

Figure 7 shows more visualization results of our method on Xia dataset, the results demonstrate the significant performance of our approach.

Content embedding Figure 8 presents the content embedding of our model. Each point is colored by its style label or content label. From the figure, we can observe that there is no clear boundary to distinguish these clusters in (a), suggesting that style information has been removed; however, once we are given motions that cover identity style and multiple categories of content, our method can clearly distinguish them in (b). This demonstrates that the style information may have been removed in both situations, which consolidates our argument.

Quantitative evaluation To better observe the transferred results in quantitative analysis, we conduct experiments on two different datasets (CMU dataset and Xia dataset), respectively. The evaluation results are presented in Table 1. We use the bold to highlight the minimum value of each column. As seen in the table, our self-supervised approach achieves comparable results when compared with the state-of-the-art methods. Table 2 shows the scores of FMD, compared with these methods, our model achieves the lowest FMD scores and is capable of generating superior motion results.

Table 3 shows the evaluation results of our method and other SOTA approaches on the KID and Diversity score. From this table, our model also gains the best performance on the KID score, and comparable results on the Diversity score.

Ablation studies To evaluate the effectiveness of our proposed contrastive learning modules, we also conduct ablative analyses on the datasets.

In our loss function, the super-parameters β and γ control the impact of content and style separation. Theoretically, once we increase the parameter β, the errors of the content distances will be reduced, but will increase the errors of the style distances. Table 1 presents the results, as we can see from the table, when we higher the value of β, the values of content distance decrease with the style distance increases in all the experimented data classes, which validates our argument. After many experiments, we find that the weights of β and γ between 0.3 and 0.5 are the most reasonable. Thus we set $\alpha = 1$, $\beta = 0.3$ and $\gamma = 0.3$ as our baseline setting, in order to balance between the content and style preservation.

Table 1 We make the quantitative comparisons of our method and others methods on the CMU dataset

Method	Zombie		Shy		Injured (unseen)		Elated (unseen)	
	Content Distance	Style Distance	Content Distance	Style Distance	Content Distance	Style Distance	Content Distance	Style Distance
Aberman	0.0344	0.2160	0.0197	0.4072	0.0217	**0.5123**	0.209	**2.2209**
Pan	0.0278	**0.2152**	0.0151	**0.1194**	0.1028	0.6783	0.0895	3.5597
Our Method (0.3)	**0.0235**	0.2215	**0.0139**	0.3315	**0.0152**	0.5235	**0.0195**	2.2391
Our Method (0)	0.0389	0.2132	0.0359	0.3159	0.1092	0.5132	0.1995	2.2219
Our Method (0.6)	0.0196	0.2465	0.0112	0.3932	0.0122	0.5645	0.0174	2.7679
Our Method (1.0)	0.0159	0.2955	0.0101	0.4269	0.0111	0.6123	0.0121	2.8952

In the upper part of the table, the bold highlights the minimum value of each column. In the lower part of the table, we show the ablation study results on the parameter β, where the number in brackets denotes the value of β with λ=0.3 in the setting

Table 2 Quantitative evaluations of our method and other recent style transfer methods on the Xia dataset, and ablation study results about loss function. The bold number indicates the best performance

Method	FMD↓	Ablation study	FMD↓
Aberman et al. [4]	30.83	$\mathcal{L}_{rec}$+$\mathcal{L}_{gan}$	9.216
Park et al. [7]	9.9	$\mathcal{L}_{rec}$+$\mathcal{L}_{gan}$+$\mathcal{L}_{nce}$	10.539
Tao et al. [46]	4.8	$\mathcal{L}_{rec}$+$\mathcal{L}_{gan}$+$\mathcal{L}_{tri}$	10.299
Ours	**4.7**	$\mathcal{L}_{total}$	6.64

Table 3 Metrics of KID and diversity on the Xia dataset, where the bold numbers indicate the best performance

Method	KID↓	Diversity↑
Aberman et al. [4]	$2.669^{\pm 0.513}$	**17.78**
Park et al. [7]	$1.355^{\pm 0.103}$	10.40
Ours	$\mathbf{0.682}^{\pm 0.095}$	13.43

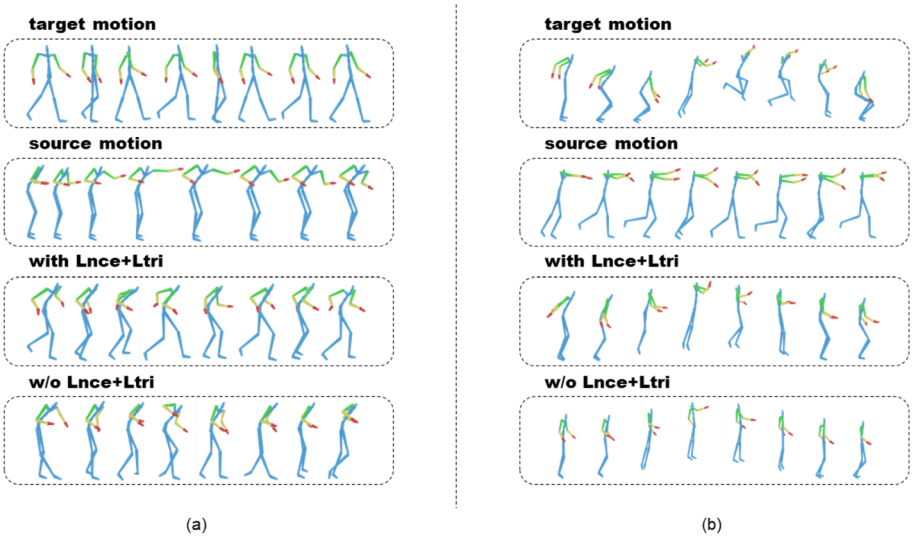
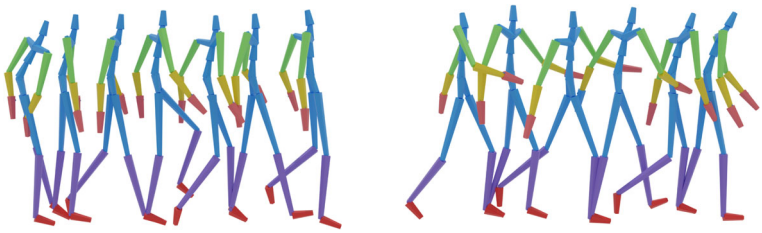


Fig. 9 Visual comparisons of our method w/ and w/o the contrastive loss elements in our loss function



(a) Depressed to happy (b) Strutting to old

Fig. 10 Style interpolation. Our style latent code facilitates to motion style interpolation

To further verify the effectiveness of multiple loss elements in our loss function, we also conduct an ablation study on Xia dataset with employing the FMD metric, mainly focusing on the ability of infoNCE loss $\mathcal{L}_{cls}$ and Tripet loss $\mathcal{L}_{tri}$. We first carry out an experiment only using reconstruction loss and adversarial loss. Then we gradually append the value of $\mathcal{L}_{cls}$ and $\mathcal{L}_{tri}$ separately to detect the impact and performance of each loss member on the whole train, and all the results are listed in Table 2. Evidently obtaining a message from the table is that discarding each loss element leads to the out-off-balance when training the model, in other words, adding $\mathcal{L}_{cls}$ and $\mathcal{L}_{tri}$ increases the performance of model by a large margin. $\mathcal{L}_{cls}$ and $\mathcal{L}_{tri}$ is used to learn the content and style codes on the aspect of the theory, respectively. Figure 9 illustrates the visualization results of our method w/ and w/o the constrastive loss elements in our loss function. From the figure, we can see that those with contrastive learning elements can achieve more reasonable results.

Style interpolation In our framework, we learn content and style codes individually, and both the latent style code and content code can be operated directly, which will benefit the style interpolation application. To test its effect, we directly adjust one style code of a branch by interpolating with another preserved style code, and then generate the interpolated motions. The visual results are showed in Fig. 10 in which we present two style interpolation effects, i.e. from depressed to happy and from strutting to old. As can be seen from these figures, our method smoothly blended different styles and successfully generated new motions.

Figure 11 further illustrates the interpolation results, where we interpolate the style embedding and time separately to better show the details. From the results, our method can achieve natural and coherent outcomes consistently.

4.4 User study

Since style has no clear definition, it is a term containing subjective factors. Thus, similar to previous work, we also conduct a user study to evaluate the results on the XIA dataset. Our questionnaire borrows heavily from the work of [4]. Specifically, 21 questions were raised in three categories, they are:

Realism The generated motion sequences should be natural and faithfully according to human eyes. We collect part of our results, and present the results driven by different approaches. Then we ask the users to answer the questions like: "Which motion looks more like an old man's walk?"

50 volunteers, aged 20-23, 25 males and 25 females, have participated in answering these questions, all of them have experience in operating MAYA or 3DMAX. Table 4 reveals the statistical result, from which we can see that 37% of our results were judged as realistic and natural.

Content preservation and style transfer We also evaluate the quality of content preservation and style transfer by comparing with state-of-the-art methods. Specifically, we provide the users with the two input sequences which represented the content input and the style input separately, and three generated sequences which were computed by five approaches,

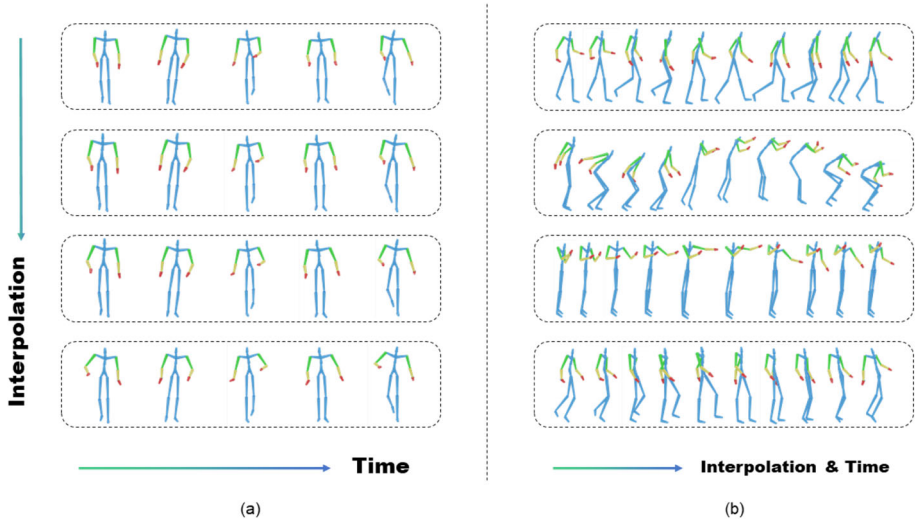


Fig. 11 Interpolation in the latent space: (a) Interpolation from depressed to angry (walk), where the style embedding and time are interpolated separately. The style weights are: 0, 0.3, 0.6, 1.0; and the time weights are: 0, 0.3, 0.5, 0.8, 1.0. (b) From top to bottom: Interpolation from angry to depressed (run), childlike to old (jump), sexy to proud (punch), neutral to strutting (kick), where the style embedding and time are consistently interpolated with weights: 0, 0.3, 0.6, 1.0. From the results, our method achieves natural and coherent motions

i.e., Holden et al. [42], Aberman et al. [4], Pan et al. [5], Tao et al. [46] and ours. The users were asked to choose the sequence that best matches the original sequence in terms of content, and similarly choose one sequence that best matches the original sequence in terms of style.

Table 4 reported the results, from which we can observe that 47% of our results were voted in maintaining the content, and 43% of our results were voted in maintaining the style.

From the user study, we can see that our method has gained much more recognition for the task of style transfer.

5 Discussion

In the following, we discuss the advantages, disadvantages, and future work of the proposed method.

In terms of advantages, although Motion Style Transfer is a critical and challenging task, our proposed framework boasts several advantages for effectively addressing this challenge. Firstly, we introduce a novel approach by disentangling the latent style code and content code.

Table 4 User study results

Categories	Holden	Aberman	Pan	Tao	Ours
Realism	9%	22%	12%	20%	37%
Content perservation	2%	25%	16%	10%	47%
Style perservation	0%	15%	19%	23%	43%

This not only enhances motion semantic learning but also allows the model to adeptly extract the necessary information. Notably, our framework stands out as the pioneer in achieving disentangled representation through unsupervised learning.

Secondly, we incorporate the InfoNCE loss and Triplet loss into our method, contributing to a more comprehensive acquisition of content and style codes in the latent space. The employment of these loss functions further refines the model's ability to discern nuanced details in motion sequences. The resulting exceptional disentanglement learning of motions empowers the generation of coherent and authentic motion sequences, showcasing the effectiveness of our approach in leveraging disentangled semantics.

In terms of disadvantages, while our framework presents promising advancements, there remain certain limitations that should be addressed in future research. Firstly, as illustrated in the diversity metric diagram Table 3, it is apparent that our generated motions exhibit less diversity when compared to prior works. The balance between quality and diversity poses a challenge in generative tasks, and our method, trained to prioritize the production of natural and authentic motions, inadvertently sacrifices some diversity in synthesis. Recognizing this trade-off is crucial, and future efforts could explore strategies to enhance diversity without compromising the realism and authenticity of the generated motions.

Secondly, our current approach relies solely on a single modality, i.e. motion sequences, as the reference for generating styled motions. This singular modality reference may constrain the generative model's ability to produce diverse and exceptional results. There is notorious research [4] into extracting styles from videos, opening up the possibility of exploring alternative modalities to enrich the generative process. The exploration of alternative modalities holds promise for unlocking new dimensions of creativity and diversity in our generated results.

As for future work, there are several promising directions for our future progression. First, integrating innovative models is a good attempt and the diffusion model stands out for its ability to strike a balance between quality and diversity. This could provide a potential remedy for the observed lack of diversity in our current method. Furthermore, with the ascent of large language models, the incorporation of text prompts emerges as a high-information modality. By introducing text as a reference modality, we can harness the robust semantic capabilities of language models like GPT and CLIP, thereby expanding the learning capacity of our generative model. Additionally, exploring multi-person and long-duration generation tasks presents another direction with significant potential. Successfully tackling these problems holds practical implications for real-world applications, offering an exciting prospect for advancing the effectiveness of our model in diverse scenarios.

6 Conclusion

In this work, we have presented a self-supervised contrastive learning framework for motion style transfer, where the InfoNCE loss and Triplet loss have been investigated to obtain the disentanglement of the latent codes. We validate the effectiveness of our approach over several benchmark datasets and showcase the advantages of our method by comparing it with state-of-the-art approaches.

Author Contributions Zizhao Wu conceived the presented idea. Zizhao Wu developted the theory and algorithm. Siyuan Mao and Cheng Zhang carried out the experiments. Zizhao Wu wrote the manuscript with the support from Yigang Wang and Ming Zeng.

Data Availability The dataset and sourcecode of Refs.[59,4,44] can be found as follows:
Ref. [59]: http://mocap.cs.cmu.edu/
Ref. [4]: https://github.com/DeepMotionEditing/deep-motion-editing
Ref. [44]: https://github.com/tianxintao/Online-Motion-Style-Transfer

Declarations

Conflicts of interest All authors declare that they have no conflict of interest.

References

1. Tenenbaum JB, Freeman WT (1996) Separating style and content. In: Mozer M, Jordan MI, Petsche T (eds) NIPS, pp 662–668. MIT Press, ???
2. Holden D, Habibie I, Kusajima I, Komura T (2017) Fast neural style transfer for motion data. IEEE Comput Graph Appl 37(4):42–49
3. Holden D, Saito J, Komura T, Joyce T (2015) Learning motion manifolds with convolutional autoencoders. In: SIGGRAPH Asia, pp 18–1184. ACM, ???
4. Aberman K, Weng Y, Lischinski D, Cohen-Or D, Chen B (2020) Unpaired motion style transfer from video to animation. ACM Trans Graph 39(4):64
5. Pan J, Sun H, Kong Y (2021) Fast human motion transfer based on a meta network. Inf Sci 547:367–383
6. Wang W, Xu J, Zhang L, Wang Y, Liu J (2020) Consistent video style transfer via compound regularization. In: The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI, pp 12233–12240. AAAI Press, ???
7. Park SS, Jang D-K, Lee S-H (2021) Diverse motion stylization for multiple style domains via spatial-temporal graph-based generative model. Proceedings of the ACM on computer graphics and interactive techniques 4:1–17
8. Jang D-K, Park SS, Lee S-H (2022) Motion puzzle: Arbitrary motion style transfer by body part. ACM Trans Graph (TOG) 41:1–16
9. Yan S, Xiong Y, Lin D (2018) Spatial temporal graph convolutional networks for skeleton-based action recognition. In: AAAI Conference on Artificial Intelligence
10. Kotovenko D, Sanakoyeu A, Lang S, Ommer B (2019) Content and style disentanglement for artistic style transfer. In: 2019 IEEE/CVF international conference on computer vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019, pp 4421–4430. IEEE, ???
11. Li Y, Li Y, Lu J, Shechtman E, Lee YJ, Singh KK (2022) Contrastive learning for diverse disentangled foreground generation. In: Computer vision - ECCV. Lecture notes in computer science, vol 13676, pp 334–351. Springer, ???
12. Bengio Y, Courville AC, Vincent P (2013) Representation learning: A review and new perspectives. IEEE Trans Pattern Anal Mach Intell 35(8):1798–1828
13. Kovar L, Gleicher M, Pighin FH (2002) Motion graphs. ACM Trans Graph 21(3):473–482
14. Min J, Chai J (2012) Motion graphs++: a compact generative model for semantic motion analysis and synthesis. ACM Trans Graph 31(6):153–115312
15. Safonova A, Hodgins JK (2007) Construction and optimal search of interpolated motion graphs. ACM Trans Graph 26(3):106
16. Shapiro A, Cao Y, Faloutsos P (2006) Style components. In: Gutwin C, Mann S (eds) Graphics Interface, pp 33–39
17. Grochow K, Martin SL, Hertzmann A, Popovic Z (2004) Style-based inverse kinematics. ACM Trans Graph 23(3):522–531
18. Wang JM, Fleet DJ, Hertzmann A (2008) Gaussian process dynamical models for human motion. IEEE Trans Pattern Anal Mach Intell 30(2):283–298
19. Ukita N, Kanade T (2012) Gaussian process motion graph models for smooth transitions among multiple actions. Comput Vis Image Underst 116(4):500–509
20. Zhou L, Shang L, Shum HPH, Leung H (2014) Human motion variation synthesis with multivariate gaussian processes. Comput Animat Virtual Worlds 25(3–4):303–311
21. Lau M, Bar-Joseph Z, Kuffner J (2009) Modeling spatial and temporal variation in motion data. ACM Trans Graph 28(5):171

22. Young JE, Igarashi T, Sharlin E (2008) Puppet master: Designing reactive character behavior by demonstration. In: Gross MH, James DL (eds) Eurographics/ACM SIGGRAPH symposium on computer animation, SCA, pp 183–191. Eurographics Association, ???
23. Levine S, Wang JM, Haraux A, Popovic Z, Koltun V (2012) Continuous character control with low-dimensional embeddings. ACM Trans Graph 31(4):28–12810
24. Ma, W., Xia, S., Hodgins, J.K., Yang, X., Li, C., Wang, Z.: Modeling style and variation in human motion. In: Popovic, Z., Otaduy, M.A. (eds.) Eurographics/ACM SIGGRAPH Symposium on Computer Animation, pp. 21–30 (2010)
25. Zheng Q, Wu W, Pan H, Mitra NJ, Cohen-Or D, Huang H (2021) Inferring object properties from human interaction and transferring them to new motions. Comput. Vis. Media 7(3):375–392
26. Zhou, Y., Li, Z., Xiao, S., He, C., Huang, Z., Li, H.: Auto-conditioned recurrent networks for extended complex human motion synthesis. In: International Conference on Learning Representations, ICLR. OpenReview.net, ??? (2018)
27. Martinez, J., Black, M.J., Romero, J.: On human motion prediction using recurrent neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, pp. 4674–4683. IEEE Computer Society, ??? (2017)
28. Jain, A., Zamir, A.R., Savarese, S., Saxena, A.: Structural-rnn: Deep learning on spatio-temporal graphs. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, pp. 5308–5317 (2016)
29. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A.C., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems, pp. 2672–2680 (2014)
30. Sadoughi, N., Busso, C.: Novel realizations of speech-driven head movements with generative adversarial networks. In: ICASSP, pp. 6169–6173. IEEE, ??? (2018)
31. Starke S, Zhao Y, Komura T, Zaman KA (2020) Local motion phases for learning multi-contact character movements. ACM Trans. Graph. 39(4):54
32. Wang Z, Chai J, Xia S (2021) Combining recurrent neural networks and adversarial training for human motion synthesis and control. IEEE Trans. Vis. Comput. Graph. 27(1):14–28
33. Rose C, Cohen MF, Bodenheimer B (1998) Verbs and adverbs: Multidimensional motion interpolation. IEEE Computer Graphics and Applications 18(5):32–40
34. Hoyet L, Ryall K, Zibrek K, Park H, Lee J, Hodgins JK, O'Sullivan C (2013) Evaluating the distinctiveness and attractiveness of human motions on realistic virtual bodies. ACM Trans. Graph. 32(6):204–120411
35. Kiiski, H., Hoyet, L., Cullen, B., O'Sullivan, C., Newell, F.N.: Perception and prediction of social intentions from human body motion. In: ACM Symposium on Applied Perception, p. 134. ACM, ??? (2013)
36. Smith HJ, Neff M (2017) Understanding the impact of animated gesture performance on personality perceptions. ACM Trans. Graph. 36(4):49–14912
37. Torresani, L., Hackney, P., Bregler, C.: Learning motion style synthesis from perceptual observations. In: Schölkopf, B., Platt, J.C., Hofmann, T. (eds.) Neural Information Processing Systems, pp 1393–1400 (2006)
38. Kim, H.J., Lee, S.: Perceptual characteristics by motion style category. In: Cignoni, P., Miguel, E. (eds.) Annual Conference of the European Association for Computer Graphics, pp 1–4 (2019)
39. Hsu E, Pulli K, Popovic J (2005) Style translation for human motion. ACM Trans. Graph. 24(3):1082–1089
40. Ikemoto L, Arikan O, Forsyth DA (2009) Generalizing motion edits with gaussian processes. ACM Trans. Graph. 28(1):1–1112
41. Jing Y, Yang Y, Feng Z, Ye J, Yu Y, Song M (2020) Neural style transfer: A review. IEEE Trans. Vis. Comput. Graph. 26(11):3365–3385
42. Holden D, Saito J, Komura T (2016) A deep learning framework for character motion synthesis and editing. ACM Trans. Graph. 35(4):138–113811
43. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, pp. 2414–2423. IEEE Computer Society, ??? (2016)
44. Smith HJ, Cao C, Neff M, Wang Y (2019) Efficient neural networks for real-time motion style transfer. Proc. ACM Comput. Graph. Interact. Tech. 2(2):13–11317
45. Xu, J., Xu, H., Ni, B., Yang, X., Wang, X., Darrell, T.: Hierarchical style-based networks for motion synthesis. In: ECCV. Lecture Notes in Computer Science, vol. 12356, pp. 178–194. Springer, ??? (2020)
46. Tao, T., Zhan, X., Chen, Z., van de Panne, M.: Style-erd: Responsive and coherent online motion style transfer. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 6583–6593 (2022)
47. Wen, Y.-H., Yang, Z., Fu, H., Gao, L., Sun, Y., Liu, Y.-J.: Autoregressive stylized motion synthesis with generative flow. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 13607–13607 (2021)

48. Locatello, F., Bauer, S., Lucic, M., Rätsch, G., Gelly, S., Schölkopf, B., Bachem, O.: Challenging common assumptions in the unsupervised learning of disentangled representations. In: Chaudhuri, K., Salakhutdinov, R. (eds.) ICML. Proceedings of Machine Learning Research, vol. 97, pp. 4114–4124. PMLR, ??? (2019)
49. Xue Y, Guo Y, Zhang H, Xu T, Zhang S, Huang X (2022) Deep image synthesis from intuitive user input: A review and perspectives. Comput. Vis. Media 8(1):3–31
50. Liu, Y., Wei, F., Shao, J., Sheng, L., Yan, J., Wang, X.: Exploring disentangled feature representation beyond face identification. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, pp. 2080–2089. IEEE Computer Society, ??? (2018)
51. Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., Lerchner, A.: beta-vae: Learning basic visual concepts with a constrained variational framework. In: International Conference on Learning Representations, ICLR. OpenReview.net, ??? (2017)
52. Kim, H., Mnih, A.: Disentangling by factorising. In: Dy, J.G., Krause, A. (eds.) ICML, vol. 80, pp. 2654–2663 (2018)
53. Kumar, A., Sattigeri, P., Balakrishnan, A.: Variational inference of disentangled latent concepts from unlabeled observations. CoRR abs/1711.00848 (2017)
54. Villegas, R., Yang, J., Hong, S., Lin, X., Lee, H.: Decomposing motion and content for natural video sequence prediction. In: 5th International Conference on Learning Representations, ICLR. OpenReview.net, ??? (2017)
55. Denton, E.L., Birodkar, V.: Unsupervised learning of disentangled representations from video. In: Guyon, I., von Luxburg, U., Bengio, S., Wallach, H.M., Fergus, R., Vishwanathan, S.V.N., Garnett, R. (eds.) Advances in Neural Information Processing Systems, pp. 4414–4423 (2017)
56. van den Oord, A., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. CoRR abs/1807.03748 (2018)
57. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.B.: Momentum contrast for unsupervised visual representation learning. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, pp 9726–9735. IEEE, ??? (2020)
58. Wang, X., Gupta, A.: Unsupervised learning of visual representations using videos. In: IEEE International Conference on Computer Vision, ICCV, pp. 2794–2802. IEEE Computer Society, ??? (2015)
59. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR, pp. 539–546 (2005)
60. Zhang, Y., Tang, F., Dong, W., Huang, H., Ma, C., Lee, T., Xu, C.: Domain enhanced arbitrary image style transfer via contrastive learning. In: Nandigjav, M., Mitra, N.J., Hertzmann, A. (eds.) SIGGRAPH '22, pp. 12–1128. ACM, ??? (2022)
61. Hénaff, O.J.: Data-efficient image recognition with contrastive predictive coding. In: ICML, vol. 119, pp. 4182–4192. PMLR, ??? (2020)
62. CMU : Cmu graphics lab motion capture database. http://mocap.cs.cmu.edu/ (2019)
63. Xia S, Wang C, Chai J, Hodgins JK (2015) Realtime style transfer for unlabeled heterogeneous human motion. ACM Trans. Graph. 34(4):119–111910
64. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: NIPS (2017)
65. Binkowski, M., Sutherland, D.J., Arbel, M., Gretton, A.: Demystifying MMD gans. In: 6th International Conference on Learning Representations, ICLR (2018)