MotionCLIP: Exposing Human Motion Generation to CLIP Space

GUY TEVET*, Tel Aviv University, Israel
BRIAN GORDON*, Tel Aviv University, Israel
AMIR HERTZ, Tel Aviv University, Israel
AMIT H. BERMANO, Tel Aviv University, Israel
DANIEL COHEN-OR, Tel Aviv University, Israel

"Usain Bolt"

"Swan lake"
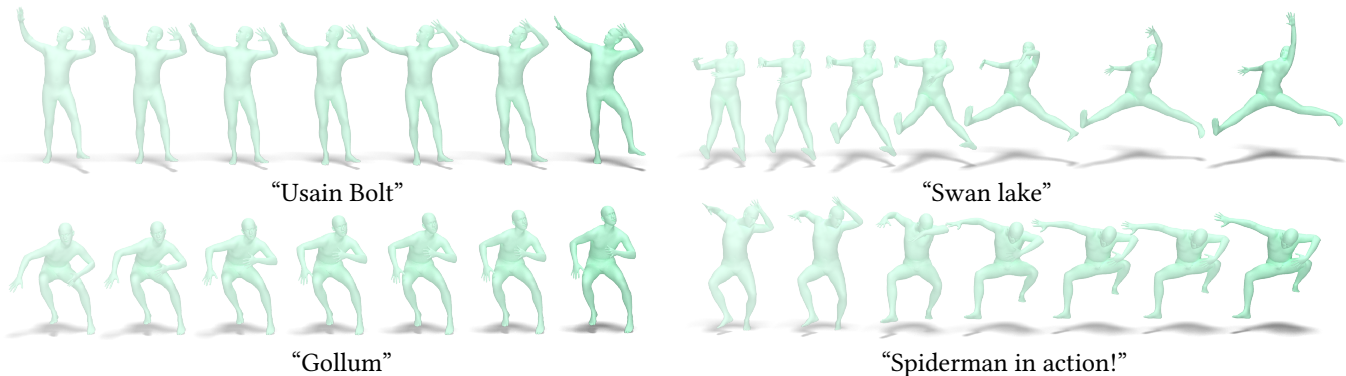
"Gollum"

"Spiderman in action!"

Fig. 1. Motions generated by MotionCLIP conditioned on different cultural references. MotionCLIP exploits the rich knowledge encapsulated in pre-trained language-images model (CLIP) and projects the human motion manifold over its latent space.

We introduce MotionCLIP, a 3D human motion auto-encoder featuring a latent embedding that is disentangled, well behaved, and supports highly semantic textual descriptions. MotionCLIP gains its unique power by aligning its latent space with that of the Contrastive Language-Image Pre-training (CLIP) model. Aligning the human motion manifold to CLIP space implicitly infuses the extremely rich semantic knowledge of CLIP into the manifold. In particular, it helps continuity by placing semantically similar motions close to one another, and disentanglement, which is inherited from the CLIP-space structure. MotionCLIP comprises a transformer-based motion auto-encoder, trained to reconstruct motion while being aligned to its text label's position in CLIP-space. We further leverage CLIP's unique visual understanding and inject an even stronger signal through aligning motion to rendered frames in a self-supervised manner. We show that although CLIP has never seen the motion domain, MotionCLIP offers unprecedented text-to-motion abilities, allowing out-of-domain actions, disentangled editing, and abstract language specification. For example, the text prompt "couch" is decoded into a sitting down motion, due to lingual similarity, and the prompt "Spiderman" results in a web-swinging-like solution that is far from seen during training. In addition, we show how the introduced latent space can be leveraged for motion interpolation, editing and recognition. [1]

1 INTRODUCTION

Human motion generation includes the intuitive description, editing, and generation of 3D sequences of human poses. It is relevant to many applications that require virtual or robotic characters. Motion generation is, however, a challenging task. Perhaps the most challenging aspect is the limited availability of data, which is expensive to acquire and to label. Recent years have brought larger sets of motion capture acquisitions [Mahmood et al. 2019], sometimes sorted by classes [Liu et al. 2019; Ji et al. 2018] or even labeled with

free text [Punnakkal et al. 2021; Plappert et al. 2016]. Yet, it seems that while this data may span a significant part of human motion, it is not enough for machine learning algorithms to understand the semantics of the motion manifold, and it is definitely not descriptive enough for natural language usage. Hence, neural models trained using labeled motion data [Ahuja and Morency 2019; Lin et al. 2018; Yamada et al. 2018; Petrovich et al. 2021; Maheshwari et al. 2022] do not generalize well to the full richness of the human motion manifold, nor to the natural language describing it.

In this work, we introduce MotionCLIP, a 3D motion auto-encoder that induces a latent embedding that is disentangled, well behaved, and supports highly semantic and elaborate descriptions. To this end, we employ CLIP [Radford et al. 2021], a large scale visual-textual embedding model. Our key insight is that even though CLIP has not been trained on the motion domain what-so-ever, we can inherit much of its latent space's virtue by enforcing its powerful and semantic structure onto the motion domain. To do this, we train a transformer-based [Vaswani et al. 2017] auto-encoder that is aligned to the latent space of CLIP, using existing motion textual labels. In other words, we train an encoder to find the proper embedding of an input sequence in CLIP space, and a decoder that generates the most fitting motion to a given CLIP space latent code. To further improve the alignment with CLIP-space, we also leverage CLIP's visual encoder, and synthetically render frames to guide the alignment in a self-supervised manner (see Figure 2). As we demonstrate, this step is crucial for out-of-domain generalization, since it allows finer-grained description of the motion, unattainable using text.

The merit of aligning the human motion manifold to CLIP space is two-fold: First, combining the geometric motion domain with lingual semantics benefits the semantic description of motion. As we show, this benefits tasks such as text-to-motion and motion style transfer.

---

[1] See our project page: https://guytevet.github.io/motionclip-page/
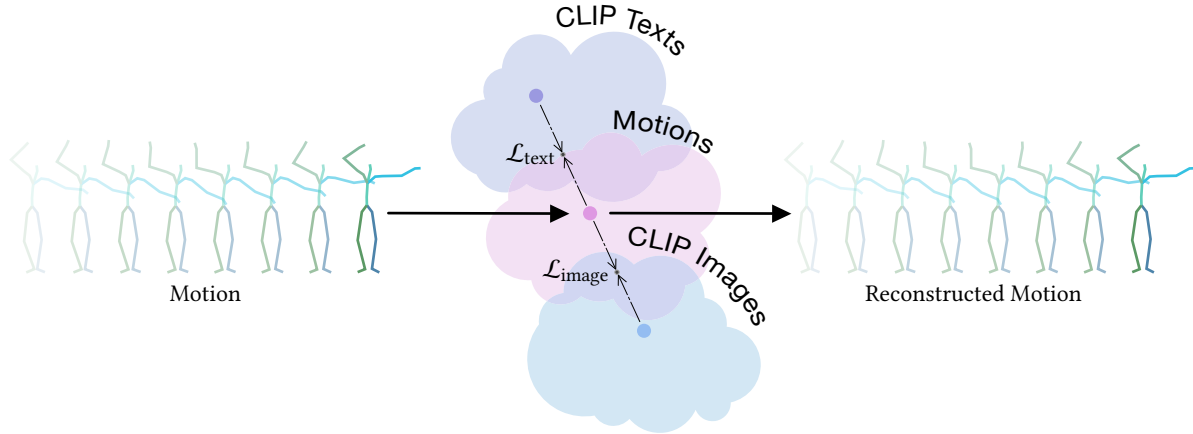
* The authors contributed equally

Fig. 2. MotionCLIP overview. A motion auto-encoder is trained to simultaneously reconstruct motion sequences while aligning their latent representation with corresponding texts and images representations in CLIP space.

More importantly however, we show that this alignment benefits the motion latent space itself, infusing it with semantic knowledge and inherited disentanglement. Indeed, our latent space demonstrates unprecedented compositionality of independent actions, semantic interpolation between actions, and even natural and linear latent-space based editing.

As mentioned above, the textual and visual CLIP encoders offer the semantic description of motion. In this aspect, our model demonstrates never-before-seen capabilities for the field of motion generation. For example, motion can be specified using arbitrary natural language, through abstract scene or intent descriptions instead of the motion directly, or even through pop-culture references. For example, the CLIP embedding for the phrase "wings" is decoded into a flapping motion like a bird, and "Williams sisters" into a tennis serve, since these terms are encoded close to motion seen during training, thanks to CLIP's semantic understanding. Through the compositionality induced by the latent space, the aforementioned process also yields clearly unseen motions, such as the iconic web-swinging gesture that is produced for the input "Spiderman" (see this and other culture references in Figure 1). Our model also naturally extents to other downstream tasks. In this aspect, we depict motion interpolation to depict latent smoothness, editing to demonstrate disentanglement, and action recognition to point out the semantic structure of our latent space. For all these applications, we show comparable or preferable results either through metrics or a user study, even though each task is compared against a method that was designed especially for it. Using the action recognition benchmark, we also justify our design choices with an ablation study.

2 RELATED WORK

2.1 Guided Human Motion Generation

One means to guide motion generation is to condition on another domain. An immediate, but limited, choice is conditioning on *action* classes. ACTOR [Petrovich et al. 2021] and Action2Motion [Guo et al. 2020] suggested learning this multi-modal distribution from existing action recognition datasets using Conditional Variational-Autoencoder(CVAE) [Sohn et al. 2015] architectures. MUGL [Maheshwari et al. 2022] model followed with elaborated Conditional Gaussian-Mixture-VAE [Dilokthanakul et al. 2016] that supports up to 120 classes and multi-person generation, based on the NTU-RGBD-120 dataset [Liu et al. 2019].

Motion can be conditioned on other domains. For example, recent works [Li et al. 2021; Aristidou et al. 2021] generated dance moves conditioned on music and the motion prefix. Edwards et al. [2016] generated facial expressions to fit a speaking audio sequence.

A more straightforward approach to control motion is using another motion. In particular, for style transfer applications. Holden et al. [2016] suggested to code style using the latent code's Gram matrix, inspired by Gatys et al. [2016]. Aberman et al. [2020] injected style attributes using a dedicated temporal-invariant AdaIN layer [Huang and Belongie 2017]. Recently, Wen et al. [2021] encoded style in the latent code of Normalizing Flow generative model [Dinh et al. 2014]. We show that MotionCLIP also encodes style in its latent representation, without making any preliminary assumptions or using a dedicated architecture.

2.2 Text-to-Motion

The KIT dataset[Plappert et al. 2016] provides about 11 hours of motion capture sequences, each sequence paired with a sentence explicitly describing the action performed. KIT sentences describe the action type, direction and sometimes speed, but lacks details about the style of the motion, and not including abstract descriptions of motion. Current text-to-motion research is heavily based on KIT. Plappert et al. [2018] learned text-to-motion and motion-to-text using seq2seq RNN-based architecture. Yamada et al. [2018] learned those two mappings by simultaneously training text and motion auto-encoders while binding their latent spaces using text and motion pairs. Lin et al. [2018] further improved trajectory prediction by adding a dedicated layer. Ahuja et al. [2019] introduced JL2P model, which got improved results with respect to nuanced concepts of the text, namely velocity, trajectory and action type. They learned
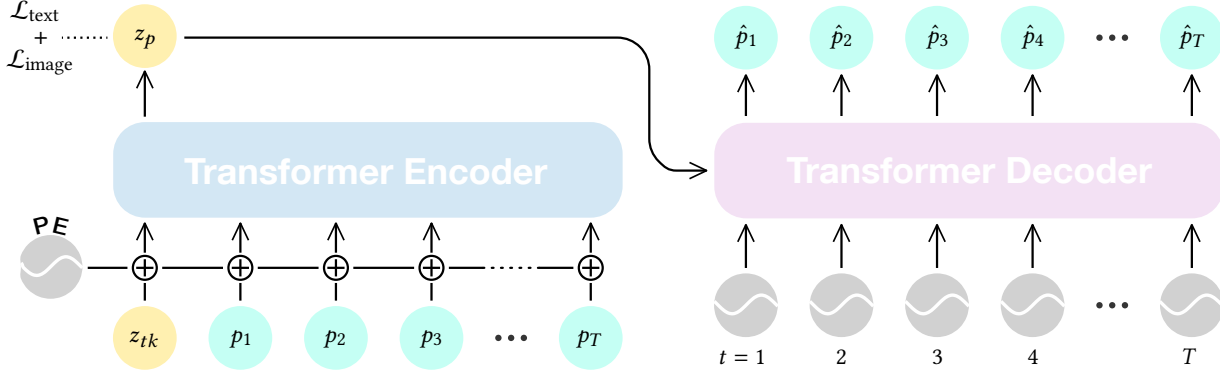
Fig. 3. Motion Auto-Encoder. A transformer encoder is trained to project a motion sequence $p_{1:T}$ into a latent vector z_p in CLIP latent space. Simultaneously, a transformer decoder is trained to recover the motion by attending to z_p.

joint motion-text latent space and apply training curriculum to ease optimization.

More recently, BABEL dataset [Punnakkal et al. 2021] provided per-frame textual labels ordered in 260 classes to the larger AMASS dataset [Mahmood et al. 2019], including about 40 hours of motion capture. Although providing explicit description of the action, often lacking any details besides the action type, this data spans a larger variety of human motion. MotionCLIP overcomes the data limitations by leveraging out-of-domain knowledge using CLIP [Radford et al. 2021].

2.3 CLIP aided Methods

Neural networks have successfully learned powerful latent representations coupling natural images with natural language describing it [He and Peng 2017; Ramesh et al. 2021]. A recent example is CLIP[Radford et al. 2021], a model coupling images and text in deep latent space using a constructive objective[Hadsell et al. 2006; Chen et al. 2020]. By training over hundred millions of images and their captions, CLIP gained a reach semantic latent representation for visual content. This expressive representation enables high quality image generation and editing, controlled by natural language [Patashnik et al. 2021; Gal et al. 2021; Frans et al. 2021]. Even more so, this model has shown that connecting the visual and textual worlds also benefits purely visual tasks [Vinker et al. 2022], simply by providing a well-behaved, semantically structured, latent space.

Closer to our method are works that utilize the richness of CLIP outside the imagery domain. In the 3D domain, CLIP's latent space provides a useful objective that enables semantic manipulation [Sanghi et al. 2021; Michel et al. 2021; Wang et al. 2021a] where the domain gap is closed by a neural rendering. CLIP is even adopted in temporal domains [Guzhov et al. 2021; Luo et al. 2021; Fang et al. 2021] that utilize large datasets of video sequences that are paired with text and audio. Unlike these works that focus on classification and retrieval, we introduce a generative approach that utilizes limited amount of human motion sequences that are paired with text.

3 METHOD

Our goal is learning a semantic and disentangled motion representation that will serve as a basis for generation and editing tasks. To this end, we need to learn not only the mapping to this representation (encoding), but also the mapping back to explicit motion (decoding).

Our training process is illustrated in Figure 2. We train a transformer-based motion auto-encoder, while aligning the latent motion manifold to CLIP joint representation. We do so using (i) a *Text Loss*, connecting motion representations to the CLIP embedding of their text labels, and (ii) an *Image Loss*, connecting motion representations to CLIP embedding of rendered images that depict the motion visually.

At inference time, semantic editing applications can be performed in latent space. For example, to perform style transfer, we find a latent vector representing the style, and simply add it to the content motion representation and decode the result back into motion. Similarly, to classify an action, we can simply encode it into the latent space, and see to which of the class text embedding it is closest. Furthermore, we use the CLIP text encoder to perform text-to-motion - An input text is decoded using the text encoder then directly decoded by our motion decoder. The implementation of these and other applications is detailed in Section 4.

We represent motion sequences using the SMPL body model [Loper et al. 2015]. A sequence of length T denoted $p_{1:T}$ such that $p_i \in \mathbb{R}^{24 \times 6}$ defines orientations in 6D representation[Zhou et al. 2019] for global body orientation and 23 SMPL joints, at the ith frame. The mesh vertices locations $v_{1:T}$ are calculated according to SMPL specifications with $\beta = 0$ and a neutral-gender body model following Petrovich et al. [Petrovich et al. 2021].

To project the motion manifold into the latent space, we learn a transformer-based auto-encoder [Vaswani et al. 2017], adapted to the motion domain [Petrovich et al. 2021; Wang et al. 2021b; Li et al. 2021]. MotionCLIP's architecture is detailed in Figure 3.

Transformer Encoder. E, Maps a motion sequence $p_{1:T}$ to its latent representation z_p. The sequence is embedded into the encoder's dimension by applying linear projection for each frame

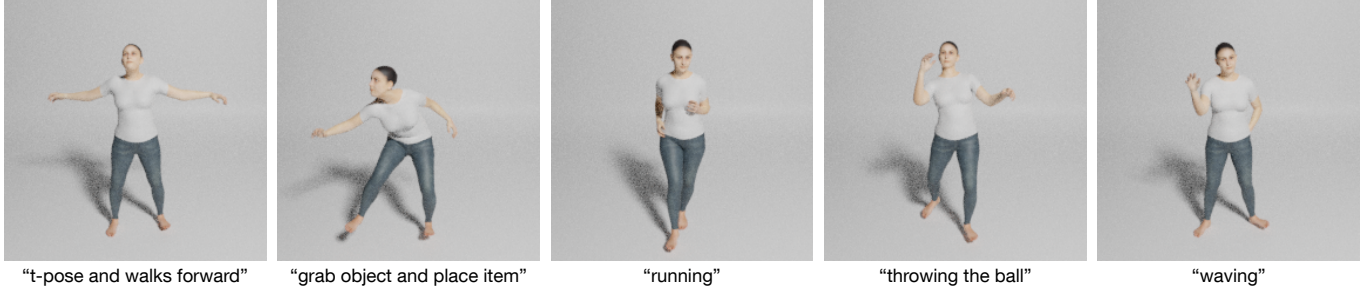| "t-pose and walks forward" | "grab object and place item" | "running" | "throwing the ball" | "waving" |

Fig. 4. A sample of the rendered frames and their text description used during training.

separately, then adding standard positional embedding. The embedded sequence is the input to the transformer encoder, together with additional learned prefix token z_{tk}. The latent representation, z_p is the first output (the rest of the sequence is dropped out). Explicitly, $z_p = E(z_{tk}, p_{1:T})$.

Transformer Decoder. D, predicts a motion sequence $\hat{p}_{1:T}$ given a latent representation z_p. This representation is fed to the transformer as key and value, while the query sequence is simply the positional encoding of $1 : T$. The transformer outputs a representation for each frame, which is then mapped to pose space using a linear projection. Explicitly, $\hat{p}_{1:T} = D(z_p)$. We further use a differentiable SMPL layer to get the mesh vertices locations, $\hat{v}_{1:T}$.

Losses. This auto-encoder is trained to represent motion via reconstruction $L2$ losses on joint orientations, joint velocities and vertices locations. Explicitly,

$$\mathcal{L}_{\text{recon}} = \frac{1}{|p|T} \sum_{i=1}^{T} \|p_i - \hat{p}_i\|^2 + \frac{1}{|v|T} \sum_{i=1}^{T} \|v_i - \hat{v}_i\|^2 \\ + \frac{1}{|p|(T-1)} \sum_{i=1}^{T-1} \|(p_{i+1} - p_i) - (\hat{p}_{i+1} - \hat{p}_i)\|^2 \tag{1}$$

Given text-motion and image-motion pairs, $(p_{1:T}, t)$, $(p_{1:T}, s)$ correspondingly, we attach the motion representation to the text and image representations using cosine distance,

$$\mathcal{L}_{\text{text}} = 1 - \cos(CLIP_{\text{text}}(t), z_p) \tag{2}$$

and

$$\mathcal{L}_{\text{image}} = 1 - \cos(CLIP_{\text{image}}(s), z_p) \tag{3}$$

The motion-text pairs can be derived from labeled motion dataset, whereas the images can be achieved by rendering a single pose from a motion sequence, to a synthetic image s, in an unsupervised manner (More details in Section 4).

Overall, the loss objective of MotionCLIP is defined,

$$\mathcal{L} = \mathcal{L}_{\text{recon}} + \lambda_{\text{text}} \mathcal{L}_{\text{text}} + \lambda_{\text{image}} \mathcal{L}_{\text{image}} \tag{4}$$

4 RESULTS

To evaluate MotionCLIP, we consider its two main advantages. In Section 4.2, we inspect MotionCLIP's ability to convert text into motion. Since the motion's latent space is aligned to that of CLIP, we use CLIP's pretrained text encoder to process input text, and convert the resulting latent embedding into motion using Motion-CLIP's decoder. We compare our results to the state-of-the-art and report clear preference for both seen and unseen generation. We also show comparable performance to state-of-the-art style transfer work simply by adding the style as a word to the text prompt. Lastly, we exploit CLIP expert lingual understanding to convert abstract text into corresponding, and sometimes unexpected, motion.

In Section 4.3 we focus on the resulting auto-encoder, and the properties of its latent-space. We inspect its smoothness and disentanglement. Smoothness is shown through well-behaved interpolations, even between distant motion. Disentanglement is demonstrated using latent space arithmetic; by adding and subtracting various motion embeddings, we achieve compositionality and semantic editing. Lastly, we leverage our latent structure to perform action recognition over the trained encoder. The latter setting is also used for ablation study. In the following, we first lay out the data used, and other general settings.

4.1 General Settings

We train our model on the BABEL dataset [Punnakkal et al. 2021]. It comprises about 40 hours of motion capture data, represented with the SMPL body model [Loper et al. 2015]. Each frame is annotated with per-frame textual labels, and is categorized into one of 260 action classes. We down sample the data to 30 frames per-second and cut it into sequences of length 60. We get a single textual label per sequence by listing all actions in a given sequence, then concatenating them to a single string. Finally, we choose for each motion sequence a random frame to be rendered using the *Blender* software and the SMPL-X add-on [Pavlakos et al. 2019] (See Figure 4). This process outputs triplets of (motion, text, synthetic image) which are used for training.

We train a transformer auto-encoder with 8 layers for each encoder and decoder as described in Section 3. We align it with the *CLIP-ViT-B/32* frozen model. Out of the data triplets, the text-motion pairs are used for the *text loss* and image-motion pairs for the *image loss*. Both λ values are set to 0.01 throughout our experiments. [2]
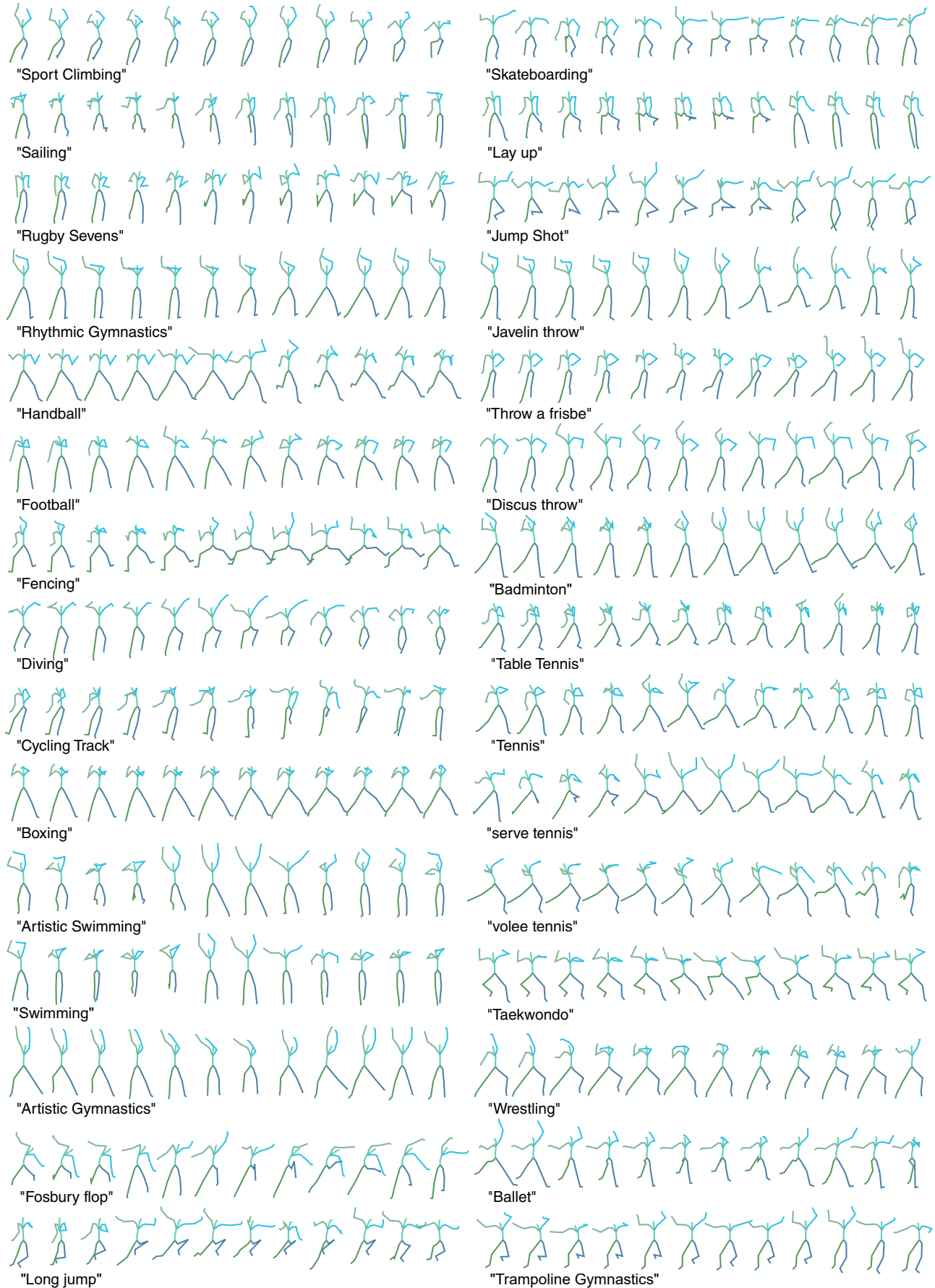
---

[2]https://github.com/GuyTevet/MotionCLIP

"Sport Climbing"

"Sailing"

"Rugby Sevens"

"Rhythmic Gymnastics"

"Handball"

"Football"

"Fencing"

"Diving"

"Cycling Track"

"Boxing"

"Artistic Swimming"

"Swimming"

"Artistic Gymnastics"

"Fosbury flop"

"Long jump"

"Skateboarding"

"Lay up"

"Jump Shot"

"Javelin throw"

"Throw a frisbe"

"Discus throw"

"Badminton"

"Table Tennis"

"Tennis"

"serve tennis"

"volee tennis"

"Taekwondo"

"Wrestling"

"Ballet"

"Trampoline Gymnastics"

Fig. 5. Sport motions generated by MotionCLIP conditioned on the text beneath each row.

	JL2P [2019]		MotionCLIP	
	seen in train	pref	seen in train	pref
In-domain set	yes	23.3%	yes	**76.7%**
Out-of-domain set	no	24.7%	no	**75.3%**

Table 1. Action generation from text - user study. *pref* is the preference score of each model (when compared side-by-side). *Seen in train* notes whether or not the samples are taken from a distribution seen by each model during train. MotionCLIP is clearly preferred by the users.

	Aberman et al. [2020]	MotionCLIP
Happy	31.3%	**68.7%**
Proud	**86.4%**	13.6%
Angry	43.5%	**56.5%**
Childlike	**57.6%**	42.4%
Depressed	**74.2%**	25.8%
Drunk	**50%**	**50%**
Old	**57.7%**	42.3%
Heavy	**85.2%**	14.8%
Average	**62.1%**	37.9%

Table 2. Style generation - user study (preference score side-by-side). We compare our style + action generation from text, to those of Aberman et al. [Aberman et al. 2020] which gets style and content motions as input. Interestingly, although not trained to generate style, our model wins twice and break even once

4.2 Text-to-Motion

Text-to-motion is performed at inference time, using the CLIP text encoder and MotionCLIP decoder, without any further training. Even though not directly trained for this task, MotionCLIP shows unprecedented performance in text-to-motion, dealing with explicit descriptions, subtle nuances and abstract language.

Actions. We start by demonstrating the capabilities of Motion-CLIP to generate explicit actions - both seen and unseen in training. We compare our model to JL2P [Ahuja and Morency 2019]. Since the two models were trained on different datasets, we define a new common ground for evaluation. We define two new sets of samples for a user study: (1)The *in-domain set* comprises actions with textual labels that appear in at least 0.5% of the labels of both datasets, and (2) the *Out-of-domain set* includes textual labels that do not appear in any of the labels of both datasets, hence, unseen for both models. For fairness, we construct this set from the list of Olympic sports (both summer and winter) that are disjoint to both datasets. We conduct a user study, comparing the generation of each model conditioned on a given textual label. For each example, we then ask users to choose which of the two motions best fits the label. Table 1 shows that MotionCLIP was clearly preferred by the users for both sets. Figure 5 demonstrates a variety of sports performed by MotionCLIP, as used in the user-study. Note how even though this is not a curated list, the motion created according to all 30 depicted text prompts resembles the requested actions.

Styles. We investigate MotionCLIP's ability to represent motion style, without being explicitly trained for it. We compare the results produced by MotionCLIP to the style transfer model by Aberman et al. [2020]. The latter receives two input motion sequences, one indicating content and the other style, and combines them through a dedicated architecture, explicitly trained to disentangle style and content from a single sequence. In contrast, we simply feed Motion-CLIP with the action and style textual names (e.g."walk proud"). We show to users the outputs of the two models side-by-side and ask them to choose which one presents both style and/or action better (See Figure 6). Even though Aberman et al. was trained specifically for this task and gets the actual motions as an input, rather then text, Table 2 shows comparable results for the two models, with an expected favor toward Aberman et al.. This, of course, also means that MotionCLIP allows expressing style with free text, and does not require an exemplar motion to describe it. Such novel free text style augmentations are demonstrated in Figure 7.

Abstract language. One of the most exciting capabilities of MotionCLIP is generating motion given text that doesn't explicitly describe motion. This includes obvious linguistic connections, such as the act of sitting down, produced from the input text "couch". Other, more surprising examples include mimicking the signature moves of famous real and fictional figures, like *Usain Bolt* and *The Karate Kid*, and other cultural references like the famous ballet performance of *Swan Lake* and the *YMCA* dance (Figures 1 and 8). These results include motions definitely not seen during training (e.g., Spiderman in Figure 1), which strongly indicates how well the motion manifold is aligned to CLIP space.

4.3 Motion Manifold Applications

It is already well established that the CLIP space is smooth and expressive. We demonstrate its merits also exist in the aligned motion manifold, through the following experiments.

Interpolation As can be seen in Figure 9, linear interpolation between two latent codes yields semantic transitions between motions in both time and space. This is a strong indication to the smoothness of this representation. Here, the source and target motions (top and bottom respectively) were sampled from the validation set, and between them are three transitions evenly sampled from the linear trajectory between the two motion representations, then decoded by MotionCLIP.

Latent-Based Editing To demonstrate how disentangled and uniform MotionCLIP latent space is, we experiment with latent-space arithmetic to edit motion (see Figure 10). As can be seen, these linear operations allow motion compositionality - the upper body action can be decomposed from the lower body one, and recomposed with another lower body performance. In addition, Style can be added by simply adding the vector of the style name embedding. These two properties potentially enable intuitive and semantic editing even for novice users.

Action Recognition Finally, we further demonstrate how well our latent spaces is semantically structured. We show how combined with the CLIP text encoder, MotionCLIP encoder can be used for action recognition. We follow BABEL 60-classes benchmark and
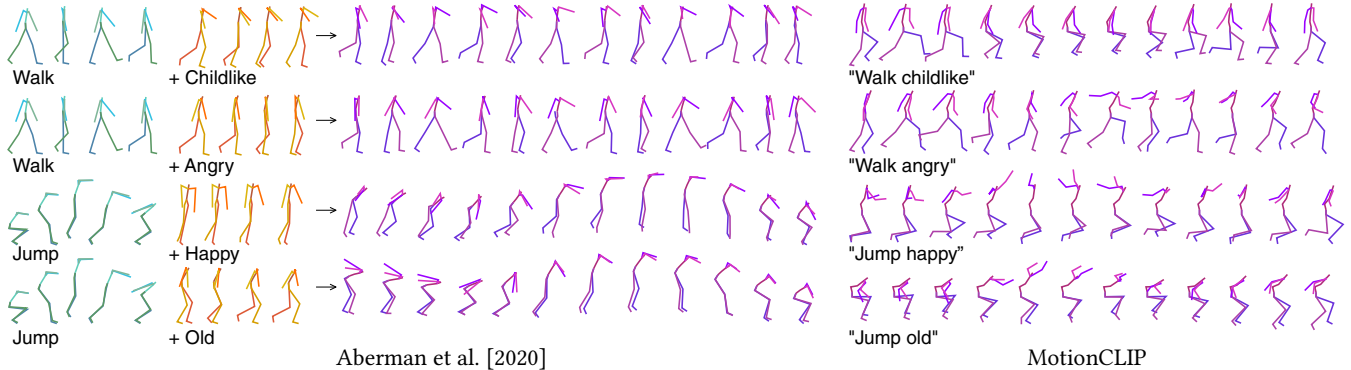
Fig. 6. Style generation. Left: style transfer by Aberman et al. [2020], conditioned on action (green) and style (orange) motions. Right: MotionCLIP generating style from plain text input.
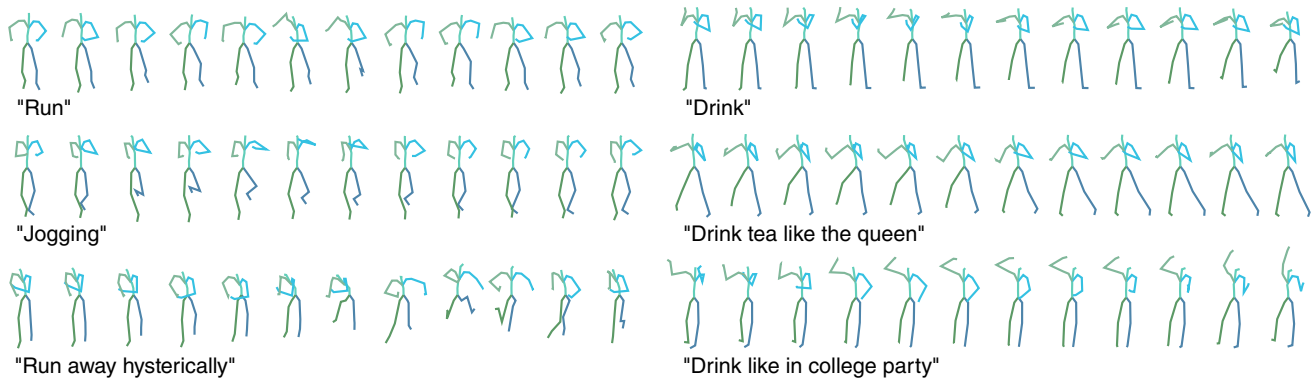

Fig. 7. MotionCLIP expresses the style described as a free text.

	Top-1 acc.	Top-5 acc.
MotionCLIP	40.9 %	57.71%
W.O. image loss	35.05%	50.26%
W.O. text loss	4.54%	18.37%
2s-AGCN [2019]	41.14%	73.18%

Table 3. Action Recognition. Using MotionCLIP together with CLIP text encoder for classification yields performance marginally close to 2s-AGCN [Shi et al. 2019] dedicated architecture on the BABEL-60 benchmark.

train the model with BABEL class names instead of the raw text. At inference, we measure the cosine distance of a given motion sequence to all 60 class name encodings and apply softmax, as suggested originally for image classification [Radford et al. 2021]. In table 3, we compare Top-1 and Top-5 accuracy of MotionCLIP classifier to 2s-AGCN classifier [Shi et al. 2019], as reported by Punnakkal et al. [2021]. As can be seen, this is another example where our framework performs similarly to dedicated state-of-the-art methods, even though MotionCLIP was not designed for it.

5 CONCLUSIONS

We have presented a motion generation network that leverages the knowledge encapsulated in CLIP, allowing intuitive operations, such as text conditioned motion generation and editing. As demonstrated, training an auto-encoder on the available motion data alone struggles to generalize well, possibly due to data quality or the complexity of the domain. Non the less, we see that the same auto-encoder with the same data can lead to a significantly better understanding of the motion manifold and its semantics, merely by aligning it to a well-behaved knowledge-rich latent space.

We restress the fascinating fact that even though CLIP has never seen anything from the motion domain, or any other temporal signal, its latent structure naturally induces semantics and disentanglement. This succeeds even though the connection between CLIP's latent space and the motion manifold is through sparse and inaccurate textual labeling. In essence, the alignment scheme transfers semantics by encouraging the encoder to place semantically similar samples closer together. Similarly, it induces the disentanglement built into the CLIP space, as can be seen, for example, in our latent-space arithmetic experiments.

Of course, MotionCLIP has its limitations, opening several novel research opportunities. It struggles to understand directions, (e.g.
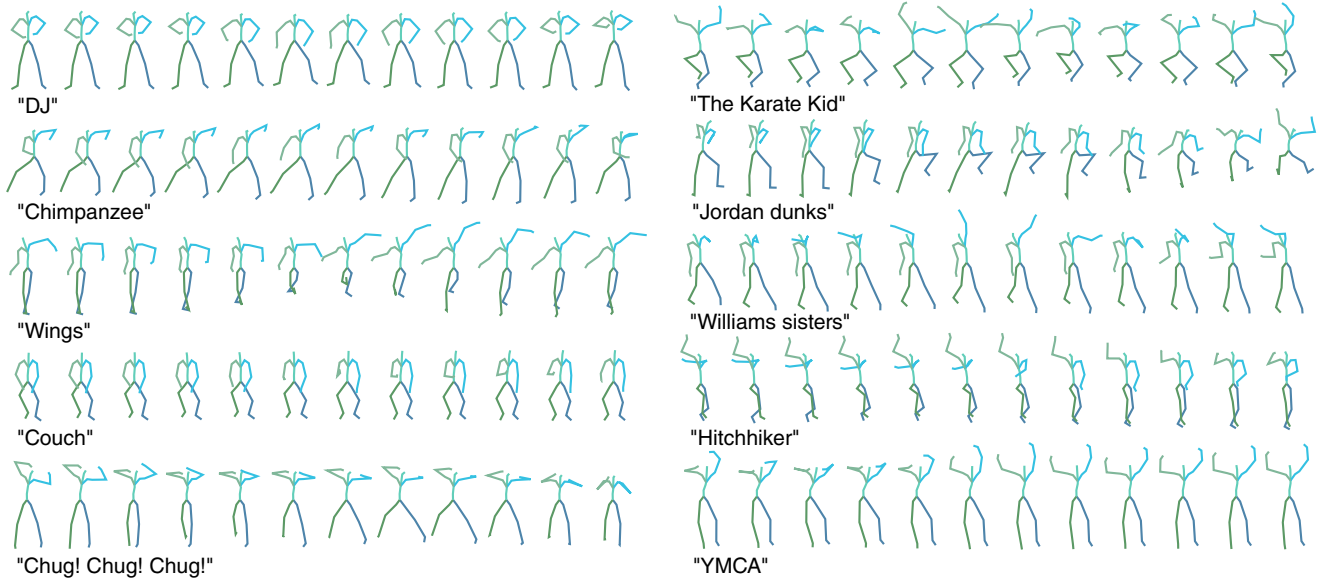
"DJ"

"Chimpanzee"

"Wings"

"Couch"

"Chug! Chug! Chug!"

"The Karate Kid"

"Jordan dunks"

"Williams sisters"

"Hitchhiker"

"YMCA"

Fig. 8. Abstract language. MotionCLIP generates the signature motions of culture figures and phrases.

Jogging

Dance Ballet

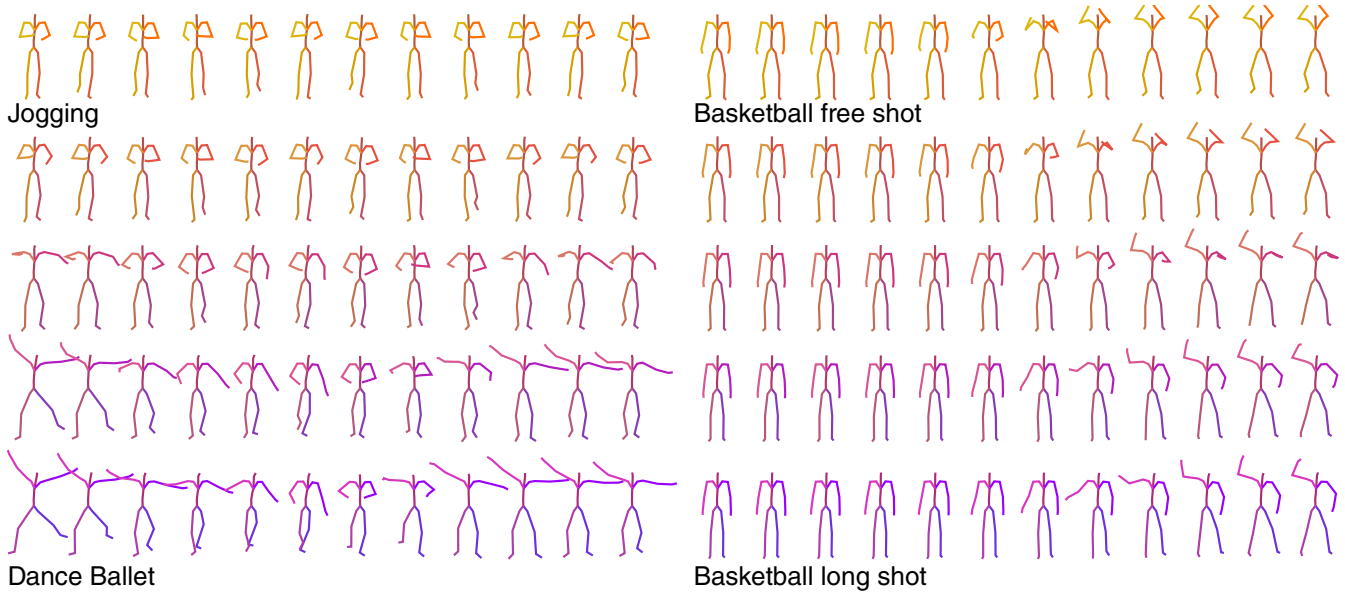Basketball free shot

Basketball long shot

Fig. 9. Latent space motion interpolation. MotionCLIP enables semantic interpolation between two motions.

left, right and counter-clockwise), to capture some styles (such as heavy and proud), and is of course not consistent for out-of-domain cultural reference exapmles (e.g, it fails to produce *Cristiano Ronaldo*'s goal celebration, and *Superman*'s signature pose). Nonetheless, we believe MotionCLIP is an important step toward intuitive motion generation. Knowledge-rich disentangled latent spaces have already proven themselves as a flexible tool to novice users in other fields, such as facial images. In the future, we would like to further explore how powerful large-scale latent spaces could

be leveraged to benefit additional domains. We would also like to explore more elaborate architectures and domain adaptation schemes for the main generation part, and to deepen our investigation into downstream tasks that could benefit from this powerful backbone.

REFERENCES

Kfir Aberman, Yijia Weng, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. 2020. Unpaired motion style transfer from video to animation. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 64–1.
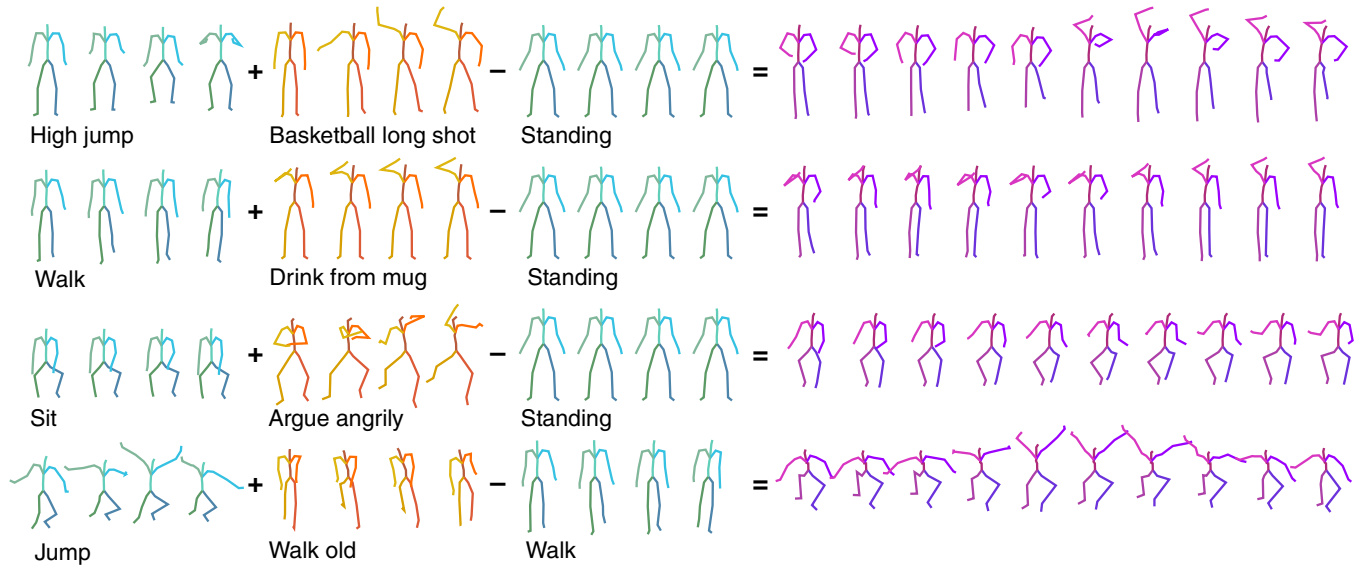
Fig. 10. Latent space motion editing. MotionCLIP enables semantic editing in latent space. Here we demonstrate two applications (1) upper and lower body action compositions (top two examples) and (2) style transfer (the two examples at the bottom).

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

Andreas Aristidou, Anastasios Yiannakidis, Kfir Aberman, Daniel Cohen-Or, Ariel Shamir, and Yiorgos Chrysanthou. 2021. Rhythm is a Dancer: Music-Driven Motion Synthesis with Global Structure. *arXiv preprint arXiv:2111.12159* (2021).

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*. PMLR, 1597–1607.

Nat Dilokthanakul, Pedro AM Mediano, Marta Garnelo, Matthew CH Lee, Hugh Salimbeni, Kai Arulkumaran, and Murray Shanahan. 2016. Deep unsupervised clustering with gaussian mixture variational autoencoders. *arXiv preprint arXiv:1611.02648* (2016).

Laurent Dinh, David Krueger, and Yoshua Bengio. 2014. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516* (2014).

Pif Edwards, Chris Landreth, Eugene Fiume, and Karan Singh. 2016. JALI: an animator-centric viseme model for expressive lip synchronization. *ACM Transactions on graphics (TOG)* 35, 4 (2016), 1–11.

Han Fang, Pengfei Xiong, Luhui Xu, and Yu Chen. 2021. CLIP2Video: Mastering Video-Text Retrieval via Image CLIP. *arXiv preprint arXiv:2106.11097* (2021).

Kevin Frans, LB Soros, and Olaf Witkowski. 2021. Clipdraw: Exploring text-to-drawing synthesis through language-image encoders. *arXiv preprint arXiv:2106.14843* (2021).

Rinon Gal, Or Patashnik, Haggai Maron, Gal Chechik, and Daniel Cohen-Or. 2021. Stylegan-nada: Clip-guided domain adaptation of image generators. *arXiv preprint arXiv:2108.00946* (2021).

Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. 2016. Image Style Transfer Using Convolutional Neural Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. 2020. Action2motion: Conditioned generation of 3d human motions. In *Proceedings of the 28th ACM International Conference on Multimedia*. 2021–2029.

Andrey Guzhov, Federico Raue, Jörn Hees, and Andreas Dengel. 2021. Audioclip: Extending clip to image, text and audio. *arXiv preprint arXiv:2106.13043* (2021).

Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, Vol. 2. IEEE, 1735–1742.

Xiangteng He and Yuxin Peng. 2017. Fine-grained image classification via combining vision and language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5994–6002.

Daniel Holden, Jun Saito, and Taku Komura. 2016. A deep learning framework for character motion synthesis and editing. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 1–11.

Xun Huang and Serge Belongie. 2017. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*. 1501–1510.

Yanli Ji, Feixiang Xu, Yang Yang, Fumin Shen, Heng Tao Shen, and Wei-Shi Zheng. 2018. A large-scale RGB-D database for arbitrary-view human action recognition. In *Proceedings of the 26th ACM international Conference on Multimedia*. 1510–1518.

Ruilong Li, Shan Yang, David A. Ross, and Angjoo Kanazawa. 2021. AI Choreographer: Music Conditioned 3D Dance Generation with AIST++. In *The IEEE International Conference on Computer Vision (ICCV)*.

Angela S Lin, Lemeng Wu, Rodolfo Corona, Kevin Tai, Qixing Huang, and Raymond J Mooney. 2018. Generating animated videos of human activities from natural language descriptions. *Learning* 2018 (2018), 1.

Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C Kot. 2019. Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. *IEEE transactions on pattern analysis and machine intelligence* 42, 10 (2019), 2684–2701.

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. 2015. SMPL: A skinned multi-person linear model. *ACM transactions on graphics (TOG)* 34, 6 (2015), 1–16.

Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. 2021. CLIP4Clip: An Empirical Study of CLIP for End to End Video Clip Retrieval. *arXiv preprint arXiv:2104.08860* (2021).

Shubh Maheshwari, Debtanu Gupta, and Ravi Kiran Sarvadevabhatla. 2022. MUGL: Large Scale Multi Person Conditional Action Generation with Locomotion. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 257–265.

Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. 2019. AMASS: Archive of Motion Capture as Surface Shapes. In *International Conference on Computer Vision*. 5442–5451.

Oscar Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. 2021. Text2Mesh: Text-Driven Neural Stylization for Meshes. *arXiv preprint arXiv:2112.03221* (2021).

Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. 2021. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2085–2094.

Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. 2019. Expressive Body Capture: 3D Hands, Face, and Body from a Single Image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 10975–10985.

Mathis Petrovich, Michael J. Black, and Gül Varol. 2021. Action-Conditioned 3D Human Motion Synthesis with Transformer VAE. In *International Conference on Computer Vision (ICCV)*. 10985–10995.

Matthias Plappert, Christian Mandery, and Tamim Asfour. 2016. The KIT motion-language dataset. *Big data* 4, 4 (2016), 236–252.

Matthias Plappert, Christian Mandery, and Tamim Asfour. 2018. Learning a bidirectional mapping between human whole-body motion and natural language using deep recurrent neural networks. *Robotics and Autonomous Systems* 109 (2018), 13–26.

Abhinanda R. Punnakkal, Arjun Chandrasekaran, Nikos Athanasiou, Alejandra Quiros-Ramirez, and Michael J. Black. 2021. BABEL: Bodies, Action and Behavior with English Labels. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*. 722–731.

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

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. 2021. Zero-shot text-to-image generation. In *International Conference on Machine Learning*. PMLR, 8821–8831.

Aditya Sanghi, Hang Chu, Joseph G Lambourne, Ye Wang, Chin-Yi Cheng, and Marco Fumero. 2021. Clip-forge: Towards zero-shot text-to-shape generation. *arXiv preprint arXiv:2110.02624* (2021).

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

Kihyuk Sohn, Honglak Lee, and Xinchen Yan. 2015. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems* 28 (2015).

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

Yael Vinker, Ehsan Pajouheshgar, Jessica Y Bo, Roman Christian Bachmann, Amit Haim Bermano, Daniel Cohen-Or, Amir Zamir, and Ariel Shamir. 2022. CLIPasso: Semantically-Aware Object Sketching. *arXiv preprint arXiv:2202.05822* (2022).

Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. 2021a. CLIP-NeRF: Text-and-Image Driven Manipulation of Neural Radiance Fields. *arXiv preprint arXiv:2112.05139* (2021).

Jiashun Wang, Huazhe Xu, Medhini Narasimhan, and Xiaolong Wang. 2021b. Multi-Person 3D Motion Prediction with Multi-Range Transformers. *Advances in Neural Information Processing Systems* 34 (2021).

Yu-Hui Wen, Zhipeng Yang, Hongbo Fu, Lin Gao, Yanan Sun, and Yong-Jin Liu. 2021. Autoregressive Stylized Motion Synthesis with Generative Flow. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13612–13621.

Tatsuro Yamada, Hiroyuki Matsunaga, and Tetsuya Ogata. 2018. Paired recurrent autoencoders for bidirectional translation between robot actions and linguistic descriptions. *IEEE Robotics and Automation Letters* 3, 4 (2018), 3441–3448.

Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. 2019. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5745–5753.