PoseScript: 3D Human Poses from Natural Language

Ginger Delmas[1,2], Philippe Weinzaepfel[2], Thomas Lucas[2],
Francesc Moreno-Noguer[1], and Grégory Rogez[2]

[1] Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Barcelona, Spain
[2] NAVER LABS Europe

Abstract. Natural language is leveraged in many computer vision tasks such as image captioning, cross-modal retrieval or visual question answering, to provide fine-grained semantic information. While human pose is key to human understanding, current 3D human pose datasets lack detailed language descriptions. In this work, we introduce the PoseScript dataset, which pairs a few thousand 3D human poses from AMASS with rich human-annotated descriptions of the body parts and their spatial relationships. To increase the size of this dataset to a scale compatible with typical data hungry learning algorithms, we propose an elaborate captioning process that generates automatic synthetic descriptions in natural language from given 3D keypoints. This process extracts low-level pose information – the *posecodes* – using a set of simple but generic rules on the 3D keypoints. The posecodes are then combined into higher level textual descriptions using syntactic rules. Automatic annotations substantially increase the amount of available data, and make it possible to effectively pretrain deep models for finetuning on human captions. To demonstrate the potential of annotated poses, we show applications of the PoseScript dataset to retrieval of relevant poses from large-scale datasets and to synthetic pose generation, both based on a textual pose description. Code and dataset are available at https://europe.naverlabs.com/research/computer-vision/posescript/.

1 Introduction

'*The pose has the head down, ultimately touching the floor, with the weight of the body on the palms and the feet. The arms are stretched straight forward, shoulder width apart; the feet are a foot apart, the legs are straight, and the hips are raised as high as possible.*'. The text above describes the downward dog yoga pose[3], and a reader is able to picture such a pose from this natural language description. Being able to automatically map natural language descriptions and accurate 3D human poses would open the door to a number of applications such as helping image annotation when the deployment of Motion Capture (MoCap) systems is

[3] https://en.wikipedia.org/wiki/Downward_Dog_Pose

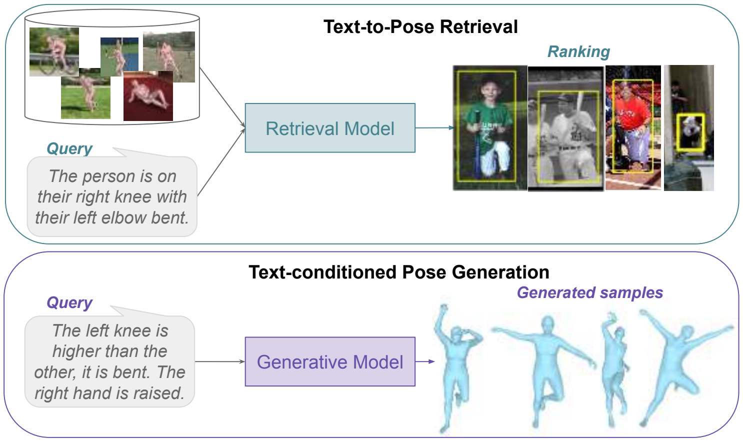
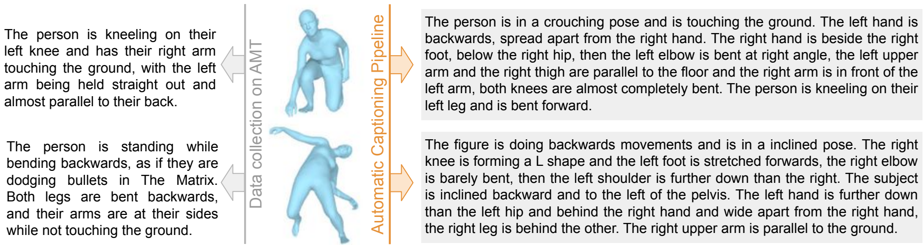

Text-to-Pose Retrieval
Ranking
Retrieval Model
Query
The person is on their right knee with their left elbow bent.
Text-conditioned Pose Generation
Generated samples
Query
The left knee is higher than the other, it is bent. The right hand is raised.
Generative Model

Fig. 1. Illustration of possible applications using PoseScript. The top figure illustrates text-to-pose retrieval where the goal is to retrieve poses in a large-scale database given a text query. This can be applied to databases of images with associated SMPL fits. The bottom figure shows an example of text-conditioned pose generation.

not practical; performing semantic searches in large-scale datasets (see Figure 1 top), which are currently only based on high-level metadata such as the action being performed [14,25,34]; complex pose or motion data generation in digital animation (see Figure 1 bottom); or teaching basic posture skills to visually impaired individuals [41].

While the problem of combining language and images or videos has attracted significant attention [17,42,20,10], in particular with the impressive results obtained by the recent multimodal neural networks CLIP [35] and DALL-E [36], the problem of linking text and 3D geometry is largely unexplored. There have been a few recent attempts at mapping text to rigid 3D shapes [8], and at using natural language for 3D object localization [7] or 3D object differentiation [1]. More recently, Fieraru *et al.* [11] introduce AIFit, an approach to automatically generate human-interpretable feedback on the difference between a reference and a target motion. There have also been a number of attempts to model humans using various forms of text. Attributes have been used for instance to model body shape [40] and face images [15]. Other approaches [12,2,30,3] leverage textual descriptions to generate motion, but without fine-grained control of the body limbs. More related to our work, Pavlakos *et al.* [28] exploit the relation between two joints along the depth dimension, and Pons-Moll *et al.* [33] describe 3D human poses through a series of *posebits*, which are binary indicators for different types of questions such as 'Is the right hand above the hips?'. However, these types of Boolean assertions have limited expressivity and remain far from the natural language descriptions a human would use.

In this paper, we propose to map 3D human poses with arbitrarily complex structural descriptions, in natural language, of the body parts and their spatial relationships. To that end, we first introduce the *PoseScript* dataset,



The person is kneeling on their left knee and has their right arm touching the ground, with the left arm being held straight out and almost parallel to their back.

The person is standing while bending backwards, as if they are dodging bullets in The Matrix. Both legs are bent backwards, and their arms are at their sides while not touching the ground.

The person is in a crouching pose and is touching the ground. The left hand is backwards, spread apart from the right hand. The right hand is beside the right foot, below the right hip, then the left elbow is bent at right angle, the left upper arm and the right thigh are parallel to the floor and the right arm is in front of the left arm, both knees are almost completely bent. The person is kneeling on their left leg and is bent forward.

The figure is doing backwards movements and is in a inclined pose. The right knee is forming a L shape and the left foot is stretched forwards, the right elbow is barely bent, then the left shoulder is further down than the right. The subject is inclined backward and to the left of the pelvis. The left hand is further down than the left hip and behind the right hand and wide apart from the right hand, the right leg is behind the other. The right upper arm is parallel to the ground.

Fig. 2. Examples of pose descriptions from PoseScript, produced by human annotators (left) and by our automatic captioning pipeline (right).

which consists of captions written by human annotators for about 4,000 poses from the AMASS dataset [25]. To scale-up this dataset, we additionally propose an automatic captioning pipeline for human-centric poses that makes it possible to annotate thousands of human poses in a few minutes. Our pipeline is built on (a) low-level information obtained via an extension of posebits [33] to finer-grained categorical relations of the different body parts (*e.g.* 'the knees are slightly/relatively/completely bent'), units that we refer to as *posecodes*, and on (b) higher-level concepts that come either from the action labels annotated by the BABEL dataset [25], or combinations of posecodes. We define rules to select and aggregate posecodes using linguistic aggregation principles, and convert them into sentences to produce textual descriptions. As a result, we are able to automatically extract human-like captions for a normalized input 3D pose. Importantly, since the process is randomized, we can generate several descriptions per pose, as different human annotators would do. We used this procedure to describe 20,000 poses extracted from the AMASS dataset. Figure 2 shows examples of human-written and automatic captions.

Using the PoseScript dataset, we propose to tackle two tasks, see Figure 1. The first is a cross-modal retrieval task where the goal is to retrieve from a database the poses that are most similar to a given text query; this can also be applied to RGB images by associating them with 3D human fits. The second task consists in generating human poses conditioned on a textual description. In both cases, our experiments demonstrate that it is beneficial to pretrain models using the automatic captions before finetuning them on real captions.
In summary, our contributions are threefold:

- We introduce the PoseScript dataset (Section 3). It associates human poses and structural descriptions in natural language, either obtained through human-written annotations or using our automatic captioning pipeline.
- We then study the task of text-to-pose retrieval (Section 4).
- We finally present the task of text-conditioned pose generation (Section 5).

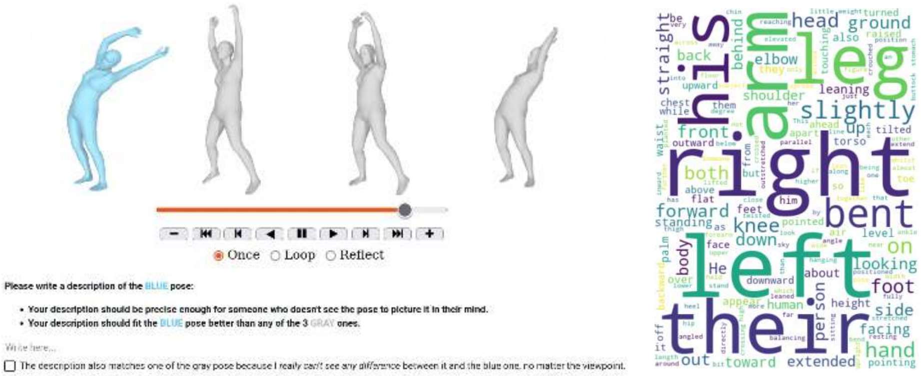
2 Related Work

Text for humans in images. Some previous works have used attributes as semantic-level representation to edit body shapes [40] or image faces [15]. In contrast, our approach focuses on body poses and leverages natural language, which has the advantage of being unconstrained and more flexible. Closer to our work, [45,6] focus on generating human 2D poses, SMPL parameters or even images from captions. However, they use MS Coco [23] captions, which are generally simple image-level statements on the activity performed by the human, and which sometimes relate to the interaction with other elements from the scene, *e.g.* 'A soccer player is running while the ball is in the air'. In contrast, we focus on fine-grained detailed captions about the pose only. FixMyPose [18] provides manually annotated captions about the difference between human poses in two synthetic images. These captions also mention objects from the environment, *e.g.* 'carpet' or 'door'. Similarly, AIFit [11] proposes to automatically generate text about the discrepancies between a reference motion and a performed one, based on differences of angles and positions. We instead focus on describing one single pose without relying on any other visual element.

Text for human motion. We deal with static poses, whereas several existing methods have mainly studied 3D action (sequence) recognition or text-based 2D [2] or 3D motion synthesis. They either condition their model on action labels [13,30,24], or descriptions in natural language [32,44,22,3,12]. Yet, even if motion descriptions effectively constrain *sequences* of poses, they do not specifically inform about individual poses. What if an animation studio looks for a sequence of 3D body poses where 'the man is running with his hands on his hips'? The model used by the artists to initialize the animation should have a deep understanding of the relations between the body parts. To this end, it is important to learn about specific pose semantics, beyond global pose sequence semantics.

Pose semantic representations. Our captioning generation process relies on posecodes that capture relevant information about the pose semantics. Posecodes are inspired from posebits [33] where images showing a human are annotated with various binary indicators. This data is used to reduce ambiguities in 3D pose estimation. Conversely, we automatically extract posecodes from normalized 3D poses in order to generate descriptions in natural language. Ordinal depth [28] can be seen as a special case of posebits, focusing on the depth relationship between two joints. They obtain annotations on some training images to improve a human mesh recovery model by adding extra constraints. Poselets [5] can also be seen as another way to extract discriminative pose information, but are not easily interpreted. In contrast to these representations, we propose to generate pose descriptions in natural language, which have the advantage (a) of being a very intuitive way to communicate ideas, and (b) of providing greater flexibility.

In summary, our proposed PoseScript dataset differs from existing datasets in that it focuses on single 3D poses instead of motion [31], and provides direct descriptions in natural language instead of simple action labels [34,13,39,21,14], binary relations [33,28] or modifying texts [18,11]. To the best of our knowledge, this is the first attempt at associating static 3D poses and descriptions in natural language.



Please write a description of the BLUE pose:

- Your description should be precise enough for someone who doesn't see the pose to picture it in their mind.
- Your description should fit the BLUE pose better than any of the 3 GRAY ones.

Write here...

□ The description also matches one of the gray pose because I really can't see any difference between it and the blue one, no matter the viewpoint.

Fig. 3. Left: Interface presented to the AMT annotators in order to collect discriminative descriptions of the blue pose. **Right: Wordcloud** of the most frequent words in the human-written descriptions.

3 The PoseScript Dataset

The PoseScript dataset is composed of static 3D human poses, together with fine-grained semantic annotations in natural language. We provide **H**uman-written annotated descriptions (PoseScript-H), and further increase the amount of data with **A**utomatically generated captions (PoseScript-A). The crowd-sourced data collection process is described in Section 3.1, and the automatic captioning pipeline in Section 3.2. Finally, aggregated statistics over the PoseScript dataset are provided in Section 3.3.

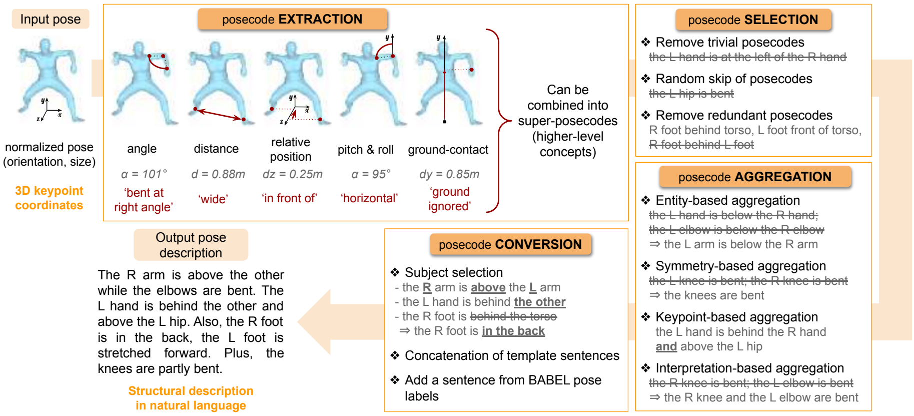
3.1 Dataset collection

We collect human-written captions for 3D human poses extracted from the AMASS dataset [25], using Amazon Mechanical Turk[4] (AMT), a crowd-sourced annotation platform. The interface, displayed in Figure 3 (left), presents the annotators with the mesh of the human pose to annotate (in blue), and a slider to control the viewpoint. To encourage discriminative captions, we additionally display 3 discriminator poses (in gray), which are semantically close to the pose to annotate. The task is to provide a description of the blue pose which is precise enough to distinguish it from the three others. We detail the discriminator selection, the complete task instructions and annotator information in the supplementary material. Some PoseScript-H examples are shown in Figure 2 (left).

3.2 Automatic captioning pipeline

We now describe the process used to generate synthetic textual descriptions for 3D human poses. As depicted in Figure 4, it relies on the extraction, selection and aggregation of elementary pieces of pose information, called *posecodes*, that are eventually converted into sentences to produce a description.

[4] https://www.mturk.com



❖ Remove trivial posecodes
~~the L hand is at the left of the R hand~~
❖ Random skip of posecodes
~~the L hip is bent~~
❖ Remove redundant posecodes
R foot behind torso, L foot front of torso, ~~R foot behind L foot~~

❖ Entity-based aggregation
~~the L hand is below the R hand;~~
~~the L elbow is below the R elbow~~
⇒ the L arm is below the R arm
❖ Symmetry-based aggregation
~~the L knee is bent; the R knee is bent~~
⇒ the knees are bent
❖ Keypoint-based aggregation
the L hand is behind the R hand **and** above the L hip
❖ Interpretation-based aggregation
~~the R knee is bent; the L elbow is bent~~
⇒ the R knee and the L elbow are bent

❖ Subject selection
- the **R** arm is **above** the **L** arm
- the L hand is behind **the other**
- the R foot is ~~behind the torso~~
⇒ the R foot is **in the back**

❖ Concatenation of template sentences
❖ Add a sentence from BABEL pose labels

The R arm is above the other while the elbows are bent. The L hand is behind the other and above the L hip. Also, the R foot is in the back, the L foot is stretched forward. Plus, the knees are partly bent.

Fig. 4. Overview of our captioning pipeline. Given a normalized 3D pose, we use posecodes to extract semantic pose information. These posecodes are then selected, merged or combined (when relevant) before being converted into a structural pose description in natural language. Letters 'L' and 'R' stand for 'left' and 'right' respectively.

The process takes 3D keypoint coordinates of human-centric poses as input. These are inferred with the SMPL-H body model [37] using the default shape coefficients and a normalized global orientation along the y-axis.

1. Posecode extraction. A posecode describes a relation between a specific set of joints. We capture five kinds of elementary relations: angles, distances and relative positions (as in [33]), but also pitch, roll and ground-contacts.

◦ *Angle posecodes* describe how a body part 'bends' at a given joint, *e.g.* the left elbow. Depending on the angle, the posecode is assigned one of the following attributes: '`straight`', '`slightly bent`', '`partially bent`', '`bent at right angle`', '`almost completely bent`' and '`completely bent`'.

◦ *Distance posecodes* categorize the $L2$-distance between two keypoints (*e.g.* the two hands) into '`close`', '`shoulder width apart`', '`spread`' or '`wide`' apart.

◦ *Posecodes on relative position* compute the difference between two keypoints along a given axis. The possible categories are, for the x-axis: '`at the right of`', '`x-ignored`', '`at the left of`'; for the y-axis: '`below`', '`y-ignored`', '`above`'; and for the z-axis: '`behind`', '`z-ignored`' and '`in front of`'. In particular, comparing the x-coordinate of the left and right hands allows to infer if they are crossed (*i.e.*, the left hand is '`at the right`' of the right hand). The '`ignored`' interpretations are ambiguous configurations which will not be described.

◦ *Pitch & roll posecodes* assess the verticality or horizontality of a body part defined by two keypoints (*e.g.* the left knee and hip together define the left thigh). A body part is '`vertical`' if it is approximately orthogonal to the y-hyperplane, and '`horizontal`' if it is in it. Other configurations are '`pitch-roll-ignored`'.

◦ *Ground-contact posecodes*, used for intermediate computation only, denote whether a keypoint is '`on the ground`' (*i.e.*, vertically close to the keypoint of minimal height in the body, considered as the ground) or '`ground-ignored`'.

Handling ambiguity in posecode categorization. Posecode categorizations are obtained using predefined thresholds. As these values are inherently subjective, we randomize the binning step by also defining a noise level applied to the measured angles and distances values before thresholding.

Higher-level concepts. We additionally define a few *super-posecodes* to extract higher-level pose concepts. These posecodes are binary (they either apply or not to a given pose configuration), and are expressed from elementary posecodes. For instance, the super-posecode '`kneeling`' can be defined as having both knees '`on the ground`' and '`completely bent`'.

2. Posecode selection aims at selecting an interesting subset of posecodes among those extracted, to obtain a concise yet discriminative description. First, we remove trivial settings (*e.g.* 'the left hand is at the left of the right hand'). Next, based on a statistical study over the whole set of poses, we randomly skip a few non-essential –*i.e.*, non-trivial but non highly discriminative – posecodes, to account for natural human oversights. We also set highly-discriminative posecodes as unskippable. Finally, we remove redundant posecodes based on statistically frequent pairs and triplets of posecodes, and transitive relations between body parts. Details are provided in the supplementary material.

3. Posecode aggregation consists in merging together posecodes that share semantic information. This reduces the size of the caption and makes it more natural. We propose four specific aggregation rules:

○ *Entity-based aggregation* merges posecodes that have similar categorizations while describing keypoints that belong to a larger entity (*e.g.* the arm or the leg). For instance 'the left hand is below the right hand' + 'the left elbow is below the right hand' is combined into 'the left arm is below the right hand'.

○ *Symmetry-based aggregation* fuses posecodes that share the same categorization, and operate on joint sets that differ only by their side of the body. The joint of interest is hence put in plural form, *e.g.* 'the left elbow is bent' + 'the right elbow is bent' becomes 'the elbows are bent'.

○ *Keypoint-based aggregation* brings together posecodes with a common keypoint. We factor the shared keypoint as the subject and concatenate the descriptions. The subject can be referred to again using *e.g.* 'it' or 'they'. For instance, 'the left elbow is above the right elbow' + 'the left elbow is close to the right shoulder' + 'the left elbow is bent' is aggregated into 'The left elbow is above the right elbow, and close to the right shoulder. It is bent.'.

○ *Interpretation-based aggregation* merges posecodes that have the same categorization, but apply on different joint sets (that may overlap). Conversely to entity-based aggregation, it does not require that the involved keypoints belong to a shared entity. For instance, 'the left knee is bent' + 'right elbow is bent' becomes 'the left knee and the right elbow are bent'.

Aggregation rules are applied at random when their conditions are met. In particular, joint-based and interpretation-based aggregation rules may operate on the same posecodes. To avoid favouring one rule over the other, merging options are first listed together and then applied at random.

4. Posecode conversion into sentences is performed in two steps. First, we select the subject of each posecode. For symmetrical posecodes – which involve two joints that only differ by their body side – the subject is chosen at random between the two keypoints, and the other is randomly referred to by its name, its side or 'the other' to avoid repetitions and provide more varied captions. For asymmetrical posecodes, we define a 'main' keypoint (chosen as subject) and 'support' keypoints, used to specify pose information (*e.g.* the 'head' in 'the left hand is raised above the head'). For the sake of flow, in some predefined cases, we omit to name the support keypoint (*e.g.* 'the left hand is raised above the head' is reduced to 'the left hand is raised'). Second, we combine all posecodes together in a final aggregation step. We obtain individual descriptions by plugging each posecode information into one template sentence, picked at random in the set of possible templates for a given posecode category. Finally, we concatenate the pieces in random order, using random pre-defined transitions. Optionally, for poses extracted from annotated sequences in BABEL [34], we add a sentence based on the associated high-level concepts (*e.g.* 'the person is in a yoga pose').

Some automatic captioning examples are presented in Figure 2 (right). The captioning process is highly modular; it allows to simply define, select and aggregate the posecodes based on different rules. Design of new kinds of posecodes (especially super-posecodes) or additional aggregation rules, can yield further improvements in the future. Importantly, randomization has been included at each step of the pipeline which makes it possible to generate different captions for the same pose, as a form of data augmentation, see supplementary material.

3.3 Dataset statistics

The PoseScript dataset contains a total of 20,000 human poses sampled from the AMASS dataset using a farthest-point sampling algorithm to maximize the variability. Specifically, we first infer the joint positions for each pose in a normalized way, using the neutral body model with the default shape coefficients and the global orientation set to 0. Then, starting from one random pose in the dataset, we iteratively select the pose with the maximum MPJE (mean per-joint error) to the set of poses that were already selected.

We collected 3.893 human annotations on AMT (PoseScript-H). We semi-automatically clean the descriptions by manually correcting the spelling of words that are not in the English dictionary, by removing one of two identical consecutive words, and by checking the error detected by a spell checker, namely NeuSpell [26]. Human-written descriptions have an average length of 55.1 tokens (51.4 words, plus punctuation). An overview of the most frequent words, among a vocabulary of 1664, is presented in Figure 3 (right).

We used the automatic captioning pipeline to increase the number of pose descriptions in the dataset (PoseScript-A). We designed a total of 87 posecodes, and automatically generated 6 captions for each of the 20,000 poses, in less than 6 minutes. Overall, automatic descriptions were produced using a posecode skipping rate of 15%, and an aggregation probability of 95%. Further details about the posecodes and other dataset statistics are provided in the supplementary.

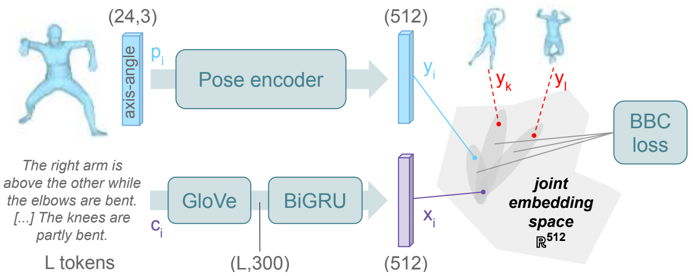

(24,3)
axis-angle
p_i
Pose encoder
(512)
y_i
y_k
y_l
BBC loss
The right arm is above the other while the elbows are bent. [...] The knees are partly bent.
L tokens
c_i
GloVe
BiGRU
(L,300)
(512)
x_i
joint embedding space ℝ^512

Fig. 5. Overview of the training scheme of the retrieval model. The input pose and caption are fed to a pose encoder and a text encoder respectively to map them into a joint embedding space. The loss encourages the pose embedding y_i and its caption embedding x_i to be close in this latent space, while being pulled apart from features of other poses in the same training batch (*e.g.* y_k and y_l).

We split the dataset into roughly 70% for training, 10% for validation and 20% for testing while ensuring that poses from the same AMASS sequence belong to the same split. When considering the automatic captions, we obtain 14,004 poses for training, 2,025 for validation and 3,971 for testing. When considering the human-written captions, each split respectively includes 2,713 (train), 400 (validation) and 780 (test) human-annotated poses.

4 Application to Text-to-Pose Retrieval

In this section, we study the problem of *text-to-pose retrieval*, which consists in ranking a large collection of poses by relevance to a given textual query (and likewise for pose-to-text retrieval). In such cross-modal retrieval task, it is standard to encode the multiple modalities into a common latent space.

Problem formulation. Let $S = \{(c_i, p_i)\}_{i=1}^{N}$ be a set of caption-and-pose pairs. By construction, p_i is the most relevant pose for caption c_i, which means that $p_{j \neq i}$ should be ranked after p_i for text-to-pose retrieval. In other words, the retrieval model aims to learn a similarity function $s(c, p) \in \mathbb{R}$ such that $s(c_i, p_i) > s(c_i, p_{j \neq i})$. As a result, a set of relevant poses can be retrieved for a given text query by computing and ranking the similarity scores between the query and each pose from the collection (the same goes for pose-to-text retrieval).

Since poses and captions are from two different modalities, we first use modality-specific encoders to embed the inputs into a joint embedding space, where the two representations will be compared to produce the similarity score.

Let $\theta(\cdot)$ and $\phi(\cdot)$ be the textual and pose encoders respectively. We denote as $x = \theta(c) \in \mathbb{R}^d$ and $y = \phi(p) \in \mathbb{R}^d$ the $L2$-normalized representations of a caption c and of a pose p in the joint embedding space (see Figure 5).

Encoders. The tokenized caption is embedded by a bi-GRU [9] taking pretrained GloVe word embeddings [29] as input. The pose is first encoded as a matrix of size $(24, 3)$, consisting of the rotation of the main 22 body joints with

	mRecall↑	pose-to-text			text-to-pose		
		R@1↑	R@5↑	R@10↑	R@1↑	R@5↑	R@10↑
test on PoseScript-A (3,971 samples)							
trained on PoseScript-A	**69.1**	**41.8**	**72.6**	**82.3**	**50.1**	**80.0**	**87.7**
test on PoseScript-H (780 samples)							
trained on PoseScript-A	7.6	2.3	9.7	13.9	1.4	6.8	11.5
trained on PoseScript-H	12.4	3.7	13.6	20.7	3.6	13.2	19.4
trained on PoseScript-A, FT on PoseScript-H	**30.4**	**11.5**	**32.1**	**42.7**	**12.6**	**35.4**	**48.0**

Table 1. Text-to-pose and pose-to-text retrieval results on the test split of the PoseScript dataset. For human-written captions (PoseScript-H), we evaluate models trained on each specific caption set alone, and one pretrained on automatic captions (PoseScript-A) then finetuned (FT) on human captions.

2 more representing the hands in axis-angle representation. The pose is then flattened and fed as input to the pose encoder, chosen as the VPoser encoder [27]: it consists of a 2-layer MLP with 512 units, batch normalization and leaky-ReLU, followed by a fully-connected layer of 32 units. We add a ReLU and a final projection layer to produce an embedding of the same size d as the text encoding.

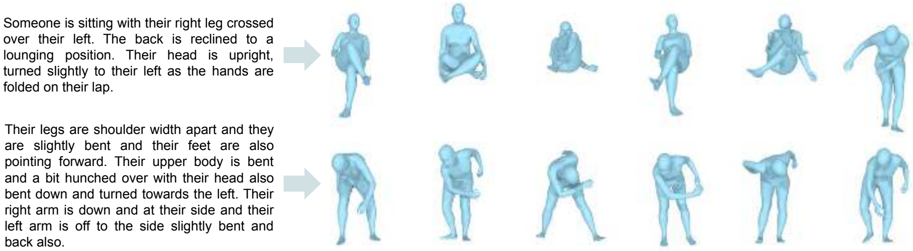
Training. Given a batch of B training pairs (x_i, y_i), we use the Batch-Based Classification (BBC) loss which is common in cross-modal retrieval [43]:

$$\mathcal{L}_{\text{BBC}} = -\frac{1}{B}\sum_{i=1}^{B} \log \frac{\exp\big(\gamma\sigma(x_i, y_i)\big)}{\sum_j \exp\big(\gamma\sigma(x_i, y_j)\big)}, \tag{1}$$

where γ is a learnable temperature parameter and σ is the cosine similarity function $\sigma(x, y) = x^\top y / \big(\|x\|_2 \times \|y\|_2\big)$.

Evaluation protocol. Text-to-pose retrieval is evaluated by ranking the whole set of poses for each of the query texts. We then compute the recall@K ($R@K$), which is the proportion of query texts for which the corresponding pose is ranked in the top-K retrieved poses. We proceed similarly to evaluate pose-to-text retrieval. We use K = 1, 5, 10 and additionally report the mean recall (mRecall) as the average over all recall@K values from both retrieval directions.

Quantitative results. We report results on the test set of PoseScript in Table 1, both on automatic and human-written captions. Our model trained on automatic captions obtains a mean recall of 69.1%, with a R@1 above 40% and a R@10 above 80% on automatic captions. However, the performance degrades on human captions, as many words from the richer human vocabulary are unseen during training on automatic captions. When trained on human captions, the model obtains a higher – but still rather low – performance. Using human captions to finetune the initial model trained on automatic ones brings an improvement of a factor 2 and more, with a mean recall (resp. R@10 for text-to-pose) of 30.4% (resp. 48.0%) compared to 12.4% (resp. 19.4%) when training from scratch. This experiment clearly shows the benefit of using the automatic captioning pipeline to scale-up the PoseScript dataset. In particular, this suggests that the



Someone is sitting with their right leg crossed over their left. The back is reclined to a lounging position. Their head is upright, turned slightly to their left as the hands are folded on their lap.

Their legs are shoulder width apart and they are slightly bent and their feet are also pointing forward. Their upper body is bent and a bit hunched over with their head also bent down and turned towards the left. Their right arm is down and at their side and their left arm is off to the side slightly bent and back also.

Fig. 6. Text-to-pose retrieval results for human-written captions from the PoseScript dataset. Directions such as 'left' and 'right' are relative to the body.

model is able to derive new concepts in human-written captions from non-trivial combination of existing posecodes in automatic captions.

Qualitative retrieval results. Examples of text-to-pose retrieval results are presented in Figure 6. It appears that the model is able to encode several pose concepts concurrently and to distinguish between the left and right body parts.

Retrieval in image databases. MS Coco [23] is one of several real-world datasets that have been used for human mesh recovery. We resort to the 74,834 pseudo-ground-truth SMPL fits provided by EFT [16], on which we apply our text-to-pose retrieval model trained with PoseScript. We then retrieve 3D poses among this MS Coco-EFT set, and display the corresponding images with the associated bounding box around the human body. Results are shown in Figure 7. We observe that overall, the constraints specified in the query text are satisfied in the images. Retrieval is based on the poses and not on the context, hence the third image of the first row where the pose is close to an actual kneeling one. This shows one application of a retrieval model trained on the PoseScript dataset: specific pose retrieval in images. Our model can be applied to any dataset of images containing humans, as long as SMPL fits are also available.

5 Application to Text-Conditioned Pose Generation

We next study the problem of *text-conditioned human pose generation*, *i.e.*, generating possible matching poses for a given text query. Our proposed model is based on Variational Auto-Encoders (VAEs) [19].

Training. Our goal is to generate a pose $\hat{p}$ given its caption c. To this end, we train a conditional VAE model that takes a tuple (p, c) composed of a pose p and its caption c at training time. Figure 8 gives an overview of our model. A pose encoder maps the pose p to a posterior over latent variables by producing the mean $\mu(p)$ and variance $\Sigma(p)$ of a normal distribution $\mathcal{N}_p = \mathcal{N}(\cdot|\mu(p), \Sigma(p))$. Another encoder is used to obtain a prior distribution $\mathcal{N}_c$, independent of p but conditioned on c. A latent variable $z \sim \mathcal{N}_p$ is sampled from $\mathcal{N}_p$ and decoded into a reconstructed pose $\hat{p}$. The training loss combines a reconstruction term $\mathcal{L}_R(p, \hat{p})$ between the original and reconstructed poses, p and $\hat{p}$ and a regularization term,

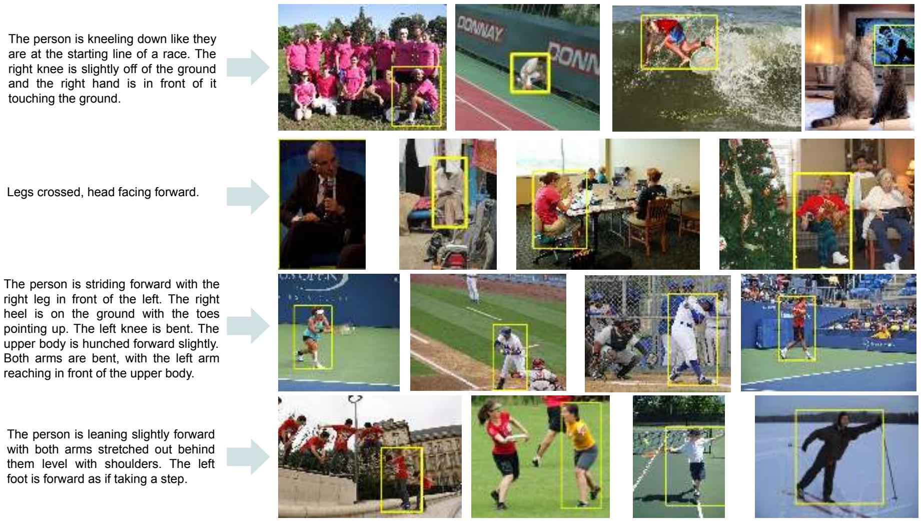

The person is kneeling down like they are at the starting line of a race. The right knee is slightly off of the ground and the right hand is in front of it touching the ground.
Legs crossed, head facing forward.
The person is striding forward with the right leg in front of the left. The right heel is on the ground with the toes pointing up. The left knee is bent. The upper body is hunched forward slightly. Both arms are bent, with the left arm reaching in front of the upper body.
The person is leaning slightly forward with both arms stretched out behind them level with shoulders. The left foot is forward as if taking a step.

Fig. 7. Retrieval results in image databases. We use our text-to-pose retrieval model trained on human captions from PoseScript to retrieve 3D poses from SMPL fits on MS Coco, for some given text queries. We display the corresponding pictures for the top retrieved poses, along with the bounding boxes around the pose.

the Kullback-Leibler (KL) divergence between $\mathcal{N}_p$ and the prior $\mathcal{N}_c$:

$$\mathcal{L} = \mathcal{L}_R(p, \hat{p}) + \mathcal{L}_{KL}(\mathcal{N}_p, \mathcal{N}_c). \tag{2}$$

We also experiment with an additional loss term, $\mathcal{L}_{KL}(\mathcal{N}_p, \mathcal{N}(\cdot|0, I))$ which is a KL divergence between the posterior and the standard Gaussian $\mathcal{N}_0 = \mathcal{N}(\cdot|0, I)$. It can be seen as another regularizer and it also allows to sample poses from the model without conditioning on captions. We treat the variance of the decoder as a learned constant [38] and use a negative log likelihood (nll) as reconstruction loss, either from a Gaussian – which corresponds to an L2 loss and a learned variance term – or a Laplacian density, which corresponds to an L1 loss. Following VPoser, we use SMPL inputs, with the axis-angle representation, and output joint rotations with the continuous 6D representation of [46]. Our reconstruction loss $\mathcal{L}_R(p, \hat{p})$ is a sum of the reconstruction losses between the rotation matrices – evaluated with a Gaussian log-likelihood – the position of the joints and the position of the vertices, both evaluated with a Laplacian log-likelihood.

Text-conditioned generation. At test time, a caption c is encoded into $\mathcal{N}_c$, from which z is sampled and decoded into a generated pose $\hat{p}$.

Evaluation metrics. We evaluate sample quality following the principle of the Fréchet inception distance: we compare the distributions of features extracted using our retrieval model (see Section 4), using real test poses and poses generated from test captions. This is denoted FID with an abuse of notation. We also report the mean-recall of retrieval models trained on real poses and evaluated on generated poses (mR R/G), and vice-versa (mR G/R). Both metrics

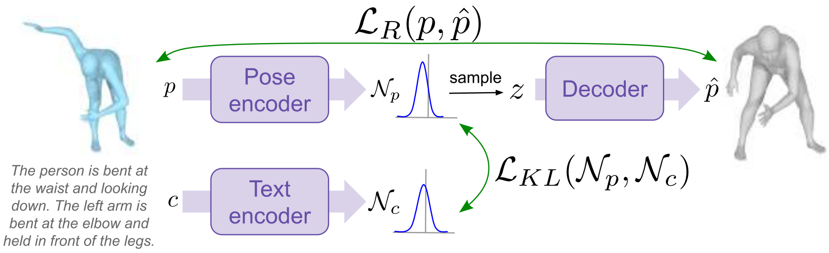

$\mathcal{L}_R(p, \hat{p})$
p
Pose encoder
$\mathcal{N}_p$
sample
z
Decoder
$\hat{p}$
$\mathcal{L}_{KL}(\mathcal{N}_p, \mathcal{N}_c)$
The person is bent at the waist and looking down. The left arm is bent at the elbow and held in front of the legs.
c
Text encoder
$\mathcal{N}_c$

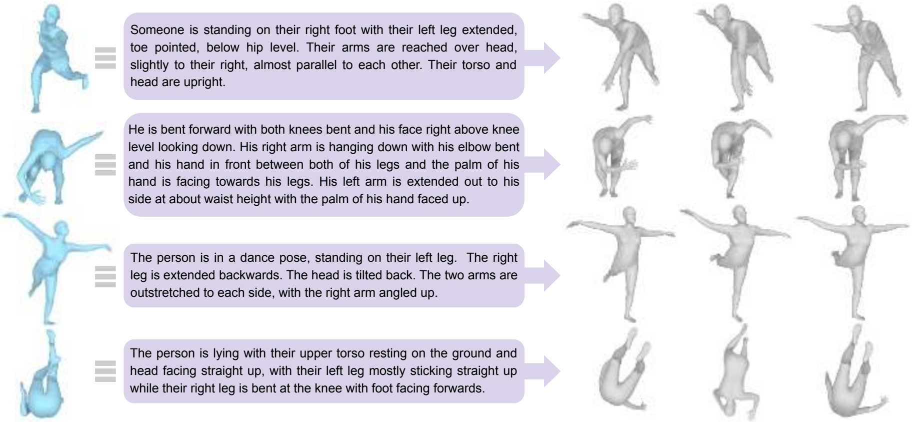
Fig. 8. Overview of the text-conditioned generative model. During training, it follows a VAE but where the latent distribution $\mathcal{N}_p$ from the pose encoder has a KL divergence term with the prior distribution $\mathcal{N}_c$ given by the text encoder. At test time, the sample z is drawn from the distribution $\mathcal{N}_c$.

	FID↓	ELBO jts↑	ELBO vert.↑	ELBO rot.↑	mRecall R/G↑	mRecall G/R↑
evaluation on automatic captions (PoseScript-A)						
without $\mathcal{L}_{KL}(\mathcal{N}_p, \mathcal{N}_0)$	0.10	1.18	1.49	0.30	24.7	14.4
with $\mathcal{L}_{KL}(\mathcal{N}_p, \mathcal{N}_0)$	**0.08**	**1.23**	**1.52**	**0.33**	**29.2**	**17.3**
evaluation on human captions (PoseScript-H) for the model with $\mathcal{L}_{KL}(\mathcal{N}_p, \mathcal{N}_0)$						
without pretraining	0.14	-0.42	0.92	-0.64	4.8	2.7
with pretraining	**0.11**	**0.50**	**1.30**	**-0.17**	**15.4**	**16.2**

Table 2. Evaluation of the text-conditioned generative model on PoseScript-A for a model without or with $\mathcal{L}_{KL}(\mathcal{N}_p, \mathcal{N}_0)$ (top) and on PoseScript-H without or with pretraining on PoseScript-A (bottom). For comparison, the mRecall when training and testing on real poses is 69.1 with PoseScript-A and 30.4 on PoseScript-H.

are sensitive to sample quality: the retrieval model will fail if the data is unrealistic. The second metric is also sensitive to diversity: missing parts of the data distribution hinder the retrieval model trained on samples. Finally, we report the Evidence Lower Bound (ELBO) computed on joints, vertices or rotation matrices, normalized by the target dimension.

Results. We present quantitative results in Table 2. We first study the impact of adding the extra-regularization loss $\mathcal{L}_{KL}(\mathcal{N}_p, \mathcal{N}_0)$ to the model trained and evaluated on automatic captions. It improves all metrics (FID, ELBO and mRecall), thus we keep this configuration and evaluate it when (a) training on human captions and (b) pretraining on automatic captions and finetuning on human captions. Pretraining improves all metrics, in particular retrieval testing and ELBOs improve substantially: pretraining helps to yield realistic and diverse samples. We display generated samples in Figure 9; the poses are realistic and generally correspond to the query. There are some variations, especially when the caption allows it, for instance with the position of the left arm in the top example or the height of the right leg in the third row. Failure cases can happen;



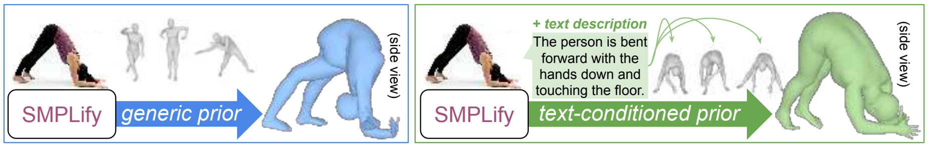
Someone is standing on their right foot with their left leg extended, toe pointed, below hip level. Their arms are reached over head, slightly to their right, almost parallel to each other. Their torso and head are upright.

He is bent forward with both knees bent and his face right above knee level looking down. His right arm is hanging down with his elbow bent and his hand in front between both of his legs and the palm of his hand is facing towards his legs. His left arm is extended out to his side at about waist height with the palm of his hand faced up.

The person is in a dance pose, standing on their left leg. The right leg is extended backwards. The head is tilted back. The two arms are outstretched to each side, with the right arm angled up.

The person is lying with their upper torso resting on the ground and head facing straight up, with their left leg mostly sticking straight up while their right leg is bent at the knee with foot facing forwards.

Fig. 9. Examples of generated samples. We show several generated samples (in grey) obtained for the human-written captions presented in the middle. For reference, we also show in blue the pose for which this annotation was originally collected.



+ text description
The person is bent forward with the hands down and touching the floor.

Fig. 10. Example of potential application to SMPL fitting in images. Using the text-conditional pose prior (right) yields a more accurate 3D pose than a generic pose prior (left) when running the optimization-based SMPL fitting method SMPLify.

in particular rare words like ‘lying’ in the bottom row lead to higher variance in the generated samples; some of them are nevertheless close to the reference.

Application to SMPL fitting in image. We showcase the potential of leveraging text data for 3D tasks on a challenging example from SMPLify [4], in Figure 10. We use our text-conditional prior instead of the generic VPoser prior [27] to initialize to a pose closer to the ground truth and to better guide the in-the-loop optimization, which helps to avoid bad local minima traps.

6 Conclusion

We introduced PoseScript, the first dataset to map 3D human poses and structural descriptions in natural language. We provided applications to text-to-pose retrieval and to text-conditioned human pose generation. For both tasks, performance is improved by pretraining on the automatic captions. Future avenues on this topic include generating images from the generated poses or exploring motion generation conditioned on complex textual description.

Acknowledgements. This work is supported in part by the Spanish government with the project MoHuCo PID2020-120049RB-I00.

References

1. Achlioptas, P., Fan, J., Hawkins, R., Goodman, N., Guibas, L.J.: Shapeglot: Learning language for shape differentiation. In: ICCV (2019)
2. Ahn, H., Ha, T., Choi, Y., Yoo, H., Oh, S.: Text2action: Generative adversarial synthesis from language to action. In: ICRA (2018)
3. Ahuja, C., Morency, L.P.: Language2pose: Natural language grounded pose forecasting. 3DV (2019)
4. Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., Black, M.J.: Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In: ECCV (2016)
5. Bourdev, L., Malik, J.: Poselets: Body part detectors trained using 3D human pose annotations. In: ICCV (2009)
6. Briq, R., Kochar, P., Gall, J.: Towards better adversarial synthesis of human images from text. arXiv preprint arXiv:2107.01869 (2021)
7. Chen, D.Z., Chang, A.X., Nießner, M.: Scanrefer: 3D object localization in rgb-d scans using natural language. In: ECCV (2020)
8. Chen, K., Choy, C.B., Savva, M., Chang, A.X., Funkhouser, T., Savarese, S.: Text2shape: Generating shapes from natural language by learning joint embeddings. In: ACCV (2018)
9. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder-decoder for statistical machine translation. In: EMNLP (2014)
10. Feng, F., Wang, X., Li, R.: Cross-modal retrieval with correspondence autoencoder. In: ACMMM (2014)
11. Fieraru, M., Zanfir, M., Pirlea, S.C., Olaru, V., Sminchisescu, C.: AIFit: Automatic 3D human-interpretable feedback models for fitness training. In: CVPR (2021)
12. Ghosh, A., Cheema, N., Oguz, C., Theobalt, C., Slusallek, P.: Synthesis of compositional animations from textual descriptions. In: ICCV (2021)
13. Guo, C., Zuo, X., Wang, S., Zou, S., Sun, Q., Deng, A., Gong, M., Cheng, L.: Action2motion: Conditioned generation of 3D human motions. In: ACMMM (2020)
14. Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments. IEEE Trans. PAMI (2014)
15. Jiang, Y., Huang, Z., Pan, X., Loy, C.C., Liu, Z.: Talk-to-edit: Fine-grained facial editing via dialog. In: ICCV (2021)
16. Joo, H., Neverova, N., Vedaldi, A.: Exemplar fine-tuning for 3D human model fitting towards in-the-wild 3D human pose estimation. In: 3DV (2020)
17. Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. In: CVPR (2015)
18. Kim, H., Zala, A., Burri, G., Bansal, M.: FixMyPose: Pose correctional captioning and retrieval. In: AAAI (2021)
19. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: ICLR (2014)
20. Li, S., Xiao, T., Li, H., Yang, W., Wang, X.: Identity-aware textual-visual matching with latent co-attention. In: ICCV (2017)
21. Li, W., Zhang, Z., Liu, Z.: Action recognition based on a bag of 3d points. In: CVPR Workshops (2010)
22. Lin, A.S., Wu, L., Corona, R., Tai, K.W.H., Huang, Q., Mooney, R.J.: Generating animated videos of human activities from natural language descriptions. In: NeurIPS workshops (2018)

23. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014)
24. Lucas, T., Baradel, F., Weinzaepfel, P., Rogez, G.: PoseGPT: Quantizing human motion for large scale generative modeling. In: ECCV (2022)
25. Mahmood, N., Ghorbani, N., Troje, N.F., Pons-Moll, G., Black, M.J.: AMASS: Archive of motion capture as surface shapes. In: ICCV (2019)
26. Muralidhar Jayanthi, S., Pruthi, D., Neubig, G.: Neuspell: A neural spelling correction toolkit. In: EMNLP (2020)
27. Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A.A., Tzionas, D., Black, M.J.: Expressive body capture: 3D hands, face, and body from a single image. In: CVPR (2019)
28. Pavlakos, G., Zhou, X., Daniilidis, K.: Ordinal depth supervision for 3D human pose estimation. In: CVPR (2018)
29. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: EMNLP (2014)
30. Petrovich, M., Black, M.J., Varol, G.: Action-conditioned 3D human motion synthesis with transformer VAE. In: ICCV (2021)
31. Plappert, M., Mandery, C., Asfour, T.: The kit motion-language dataset. Big data (2016)
32. Plappert, M., Mandery, C., Asfour, T.: Learning a bidirectional mapping between human whole-body motion and natural language using deep recurrent neural networks. Robotics Auton. Syst. (2018)
33. Pons-Moll, G., Fleet, D.J., Rosenhahn, B.: Posebits for monocular human pose estimation. In: CVPR (2014)
34. Punnakkal, A.R., Chandrasekaran, A., Athanasiou, N., Quiros-Ramirez, A., Black, M.J.: BABEL: Bodies, action and behavior with english labels. In: CVPR (2021)
35. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML (2021)
36. Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., Sutskever, I.: Zero-shot text-to-image generation. In: ICML (2021)
37. Romero, J., Tzionas, D., Black, M.J.: Embodied hands: Modeling and capturing hands and bodies together. In: SIGGRAPH Asia (2017)
38. Rybkin, O., Daniilidis, K., Levine, S.: Simple and effective vae training with calibrated decoders. In: ICML (2021)
39. Shahroudy, A., Liu, J., Ng, T.T., Wang, G.: NTU RGB+D: A large scale dataset for 3D human activity analysis. In: CVPR (2016)
40. Streuber, S., Quiros-Ramirez, M.A., Hill, M.Q., Hahn, C.A., Zuffi, S., O'Toole, A., Black, M.J.: Body talk: Crowdshaping realistic 3D avatars with words. ACM TOG (2016)
41. Suveren-Erdogan, C., Suveren, S.: Teaching of basic posture skills in visually impaired individuals and its implementation under aggravated conditions. Journal of Education and Learning (2018)
42. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: A neural image caption generator. CVPR (2015)
43. Vo, N., Jiang, L., Sun, C., Murphy, K., Li, L.J., Fei-Fei, L., Hays, J.: Composing text and image for image retrieval-an empirical odyssey. In: CVPR (2019)
44. Yamada, T., Matsunaga, H., Ogata, T.: Paired recurrent autoencoders for bidirectional translation between robot actions and linguistic descriptions. IEEE RAL (2018)

45. Zhang, Y., Briq, R., Tanke, J., Gall, J.: Adversarial synthesis of human pose from text. In: GCPR (2020)
46. Zhou, Y., Barnes, C., Lu, J., Yang, J., Li, H.: On the continuity of rotation representations in neural networks. In: CVPR (2019)