Language2Pose: Natural Language Grounded Pose Forecasting

Chaitanya Ahuja
Language Technologies Institute
Carnegie Mellon University
cahuja@andrew.cmu.edu

Louis-Philippe Morency
Language Technologies Institute
Carnegie Mellon University
morency@cs.cmu.edu

http://chahuja.com/language2pose

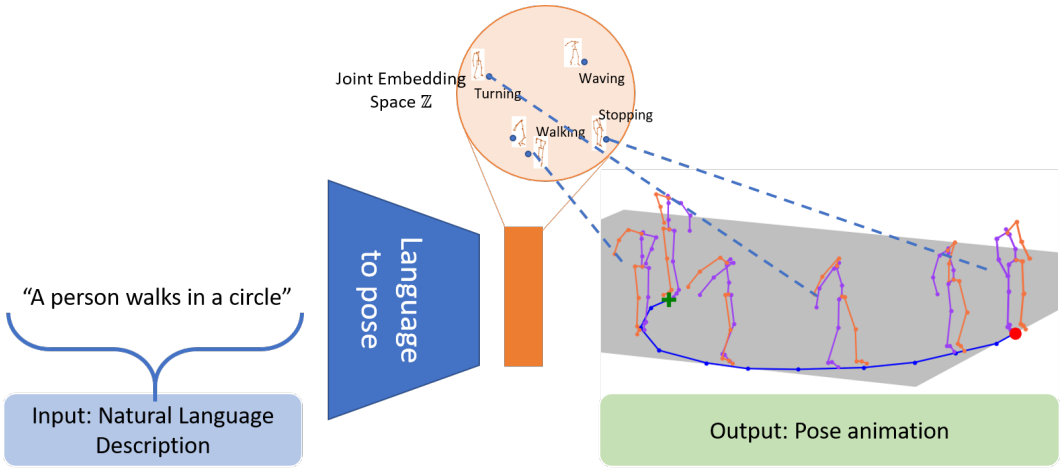

Figure 1: Overview of our model which uses joint multimodal space of language and pose to generate an animation conditioned on the input sentence.

Abstract

Generating animations from natural language sentences finds its applications in a a number of domains such as movie script visualization, virtual human animation and, robot motion planning. These sentences can describe different kinds of actions, speeds and direction of these actions, and possibly a target destination. The core modeling challenge in this language-to-pose application is how to map linguistic concepts to motion animations.

In this paper, we address this multimodal problem by introducing a neural architecture called Joint Language-to-Pose (or JL2P), which learns a joint embedding of language and pose. This joint embedding space is learned end-to-end using a curriculum learning approach which emphasizes shorter and easier sequences first before moving to longer and harder ones. We evaluate our proposed model on a publicly available corpus of 3D pose data and human-annotated sentences. Both objective metrics and human judgment evaluation confirm that our proposed approach is able to generate more accurate animations and are deemed visually more representative by humans than other data driven approaches.

1. Introduction

Generating animations from natural language descriptions is a first step for movie script visualization [11, 20] which can later be stitched together while maintaining co-references in the story-line [38]. These language grounded animations can also be useful in cases like virtual human animation [30, 7, 6], robot motion and task planning [16, 2].

An animation consists of a sequence of poses, which can be represented by positions of different joints in the body such as *Root* (base of spine), *head*, *shoulder*, *wrist*, *knee* and many more.

Pose forecasting conditioned on natural language has 3 major challenges. First, pose and natural language are very different modalities. The model needs a joint space where both natural language sentences and poses can be mapped. The model should also be able to decode animations from this embedding space. Second, different

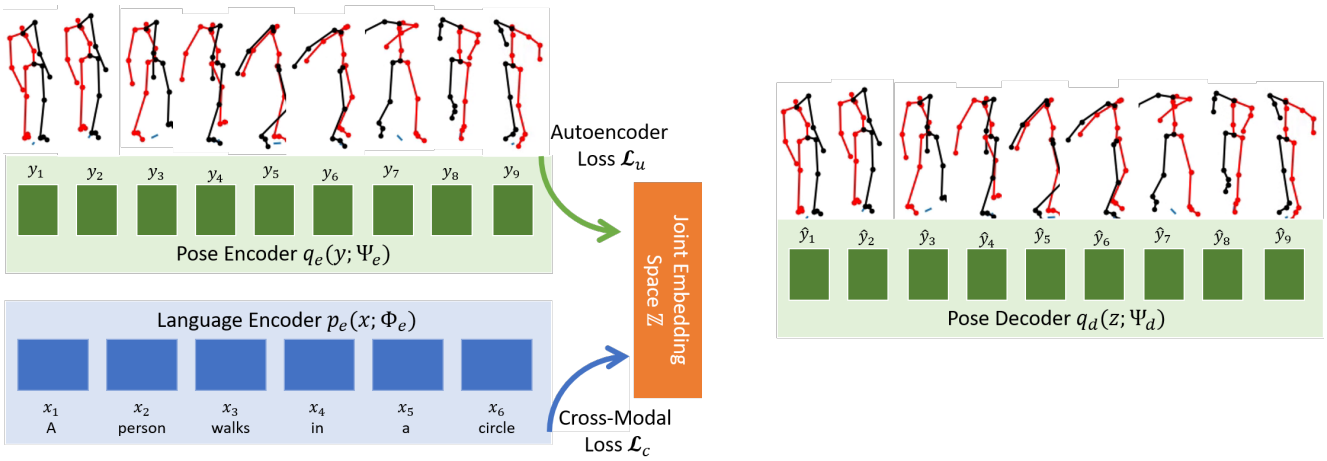

Figure 2: Overview of our proposed model Joint Language-to-Pose (or JL2P). Language and pose are mapped to a joint embedding space $\mathcal{Z}$, which can now be used by a trained pose decoder q_d to generate a pose sequence. At train time both p_e and q_e are used to create the joint embedding using a training curriculum. But at inference time $z \in \mathcal{Z}$ is encoded by p_e and decoded by q_d, giving us a model which can generate a animation (or sequence of poses) from a free form description (or language).

words of a sentence represent different qualities about the animation. Verbs and adverbs describe the action and speed/acceleration of the action; nouns and adjectives describe locations and directions respectively. The model has to map these concepts to small pose sequences and then stitch them to render convincing animations. Third, we want to see if objective metrics correlate with subjective metrics for this task as our models are trained using objective distance metrics, but the quality of generated animations can only be judged by humans.

In this paper, our two main contributions tackle the modeling challenges of pose and natural language. First, we propose a model Joint Language-to-Pose (or JL2P) that learns a joint embedding space of these two modalities. Second, we use a training curriculum to help the model emphasize more on shorter and and easier sequences first and longer and harder sequences later. Additionally, to make the training regimen robust to the outliers in the dataset, we use Smoooth L1 as the distant metric in our loss function. Through multiple objective and subjective experiments, we show that our model can generate more accurate and natural animations from natural language sentences than other data driven models.

2. Related Work

Pose Forecasting: Data driven human pose forecasting attempts to understand the behaviours of the subject from its history of poses and generates the next sequence of poses. Short-term predictions [24] focus on modeling joint angles corresponding to hands, legs, head and torso. Long-term predictions [10, 31, 24] additionally model the positions of the human character to generate animations like walking, running, jumping and crawling.

While some works use different actions (such as running, kicking, and more) as conditioning variables to generate the future pose [31, 18], others rely solely on the history of poses to predict what kind of motion will follow [8]. Pose forecasting for locomotion is a more commonly researched topic, where models decide where and when to run/walk based on low-level control parameters such as trajectory and terrain [13]. Task based locomotion (such as writing on a whiteboard, moving a box, and sitting on a box) add the nuances of transitioning from one task to another, but pose generation is based on task-specific footstep plans that act as motion templates [1].

All these approaches are either action specific, or require a set of low-level control parameters to forecast future pose. In this work, we aim replace low-level control parameters with high-level control parameters (e.g. natural language) to control actions and their speed and direction for the generated pose.

Image or Speech conditioned pose forecasting: Images with a human can act as a context to forecast what comes next. Chao et. al. [5] use one image frame to predict the next few poses. These generated poses can now be used to aid the generation of a video [35] or a sequence of images [19]. An image, a high-level control parameter, has action information for pose generation, but it does not provide a fine-grained control on the speed and acceleration of the motion trajectory.

Speech can also be used to control animations of virtual characters. Taylor et. al.[32] use a data driven approach to

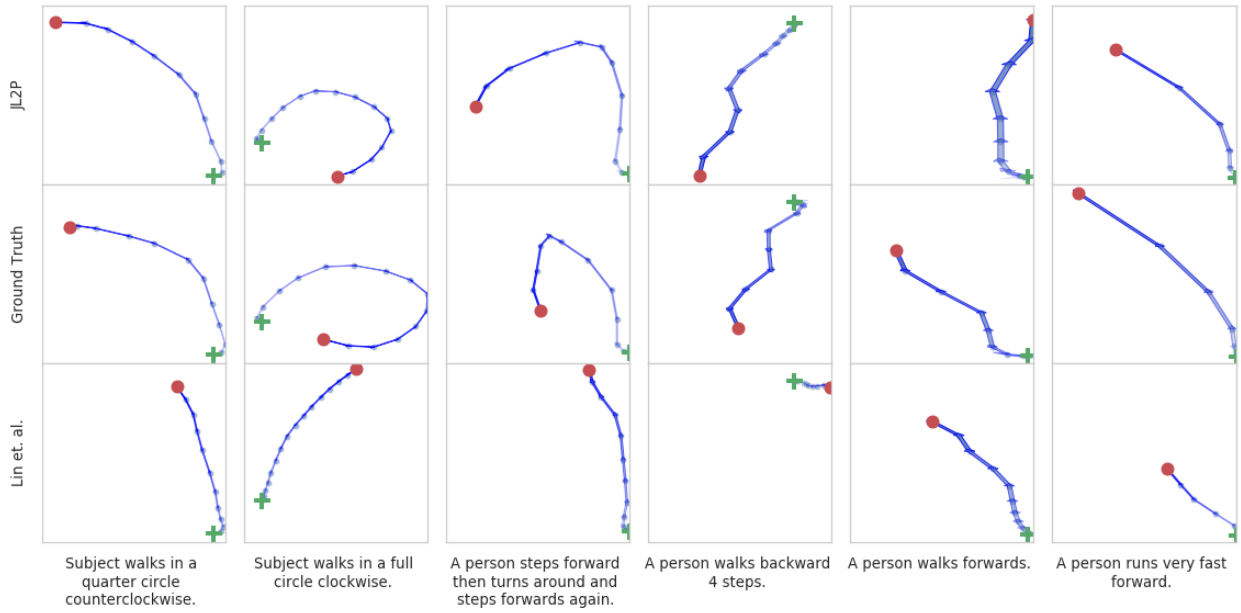

Figure 3: Trajectory plots of the generated pose (i.e. *Root*'s position) viewed from the top. Each box represents a generated trajectory of the model on the vertical axis and sentence on the horizontal axis. The person starts at the green cross (x) and ends at the red circle (•) with blue dots (•) denoting equally placed time-steps. All trajectories in each column have the same scale for fair comparison across models.

model facial animation, while upper body pose forecasting conditioned on speech inputs has been tackled by Takeuchi et. al.[30]. But, these pose sequences model the non-verbal behaviours (such as head nods, pose switches, hand waving and so on) of the character and do not offer fine-grained control over the characters next movements.

Language conditioned pose forecasting: Natural language sentences consists of verbs describing the actions, adverbs describing the speed/acceleration of the action, and nouns with adjectives to describe the direction or target. This information can help provide a more fine-grained control over pose generations compared to image or speech.

Statistical models [29, 28] which use bigram models for natural language have been trained to encode motion sequences from sentences. Ahn et. al. [2] use around 2100 hours of youtube videos with annotated text descriptions to train a pose generation model. Pose sequences extracted from videos have limited translation and occluded lower bodies, hence their model only predicts the upper body with a static *Root* joint. Some works use 3D motion capture data instead [26, 34].

Human motions generally have translation of the *Root* joint, hence forecasting trajectory is important to get natural looking animations. Lin et. al [17] generates pose of all the joints of the body by pretraining a pose2pose autoencoder model before mapping language embeddings on the learned pose space. But the embedding space is not learned jointly [23] which may limit the generative powers of the pose decoder. In contrast, our proposed approach learns a joint embedding space of language and pose using a curriculum learning training regime.

3. Problem Statement

As an example, consider a natural language sentence which describes a human's motion: *"A person walks in a circle"*. The goal of this cross-modal language-to-pose translation task is to generate an animation representing the sentence; i.e. an animation that shows a person following a trajectory of a circle with a walking motion (see figure 1).

Formally, given a sentence, represented by an N-sized sequence of words $X_{1:N} = [x_1, x_2, \dots x_N]$, we want to predict a T-sized sequence of 3D poses $Y_{1:T} = [y_1, y_2, \dots y_T]$ that are coherent with the semantics in the sentence. $x_i \in \mathcal{R}^K$ is the i^{th} word vector with dimension K. $y_t \in \mathcal{R}^{J\times 3}$ is the pose matrix at time t. Rows of y_t represent joints of the skeleton and columns are the xyz-coordinates of each joint. Tensors X and Y are elements of sets $\mathcal{X}$ and $\mathcal{Y}$ respectively.

Modeling language-to-pose is done by training a model $f : \mathcal{R}^{K\times N} \longrightarrow \mathcal{R}^{J\times 3\times T}$ to predict a pose sequence $\hat{Y}_{1:T}$

$$\hat{Y}_{1:T} = f\left(X_{1:N}; \Theta\right) \tag{1}$$

where Θ are trainable parameters of the model f.

4. Joint Language-to-Pose

Language-to-pose models should be able to grasp nuanced concepts like speed, direction of motion and the kind of actions from the language and translate them to pose sequences (or animations). This requires the model to learn a multimodal joint space of language and pose. In doing so, it should also be able to generate sequences that are deemed correlated to the sentence by humans.

To achieve that objective, we propose Joint Language-to-Pose (or JL2P) model to learn the joint embedding space. Given an input sentence, an animation can be sampled from this model at inference stage.

In this section, a joint embedding space of language and pose is formalized. This is followed by an algorithm to train for the joint embedding space and a discussion on the practical edge cases at inference time for our Joint Language-to-Pose model.

4.1. Joint Embedding Space for Language and Pose

To learn a joint embedding space of language and pose, the sentence $X_{1:N}$ and pose $Y_{1:T}$ are first mapped to a latent representation using a sentence encoder $p_e(X_{1:N}; \Phi_e)$ and a pose encoder $q_e(Y_{1:T}; \Psi_e)$ respectively. These estimate the latent representation or embeddings z_x and z_y respectively in the embedding space $\mathcal{Z} \subset \mathcal{R}^h$,

$$z_x = p_e\left(X_{1:N}; \Phi_e\right) \tag{2}$$

$$z_y = q_e\left(Y_{1:T}; \Psi_e\right) \tag{3}$$

z_x, z_y should lie close to each other in $\mathcal{Z}$ as they represent the same concept. To ensure that they do lie close together, a joint translation loss is constructed (refer to Figure 2) and trained end to end with a training curriculum.

4.2. Joint Loss Function

Once we have the embedding z_x or z_y, a pose decoder $q_d(; \Psi_d)$ is used to generate an animation from the joint embedding space $\mathcal{Z}$. The output of the pose decoder must now lie close to the pose sequence $Y_{1:T}$. Hence, using $X_{1:N}$ as inputs and $Y_{1:T}$ as outputs, the cross-modal translation loss is defined as,

$$\mathcal{L}_c = d\left(q_d\left(z_x; \Psi_d\right), Y_{1:T}\right) \tag{4}$$

and using $Y_{1:T}$ as inputs and $Y_{1:T}$ as outputs, the uni-modal translation (or autoencoder) loss is defined as,

$$\mathcal{L}_u = d\left(q_d\left(z_y; \Psi_d\right), Y_{1:T}\right) \tag{5}$$

where $d(x, y)$ is a function to calculate the distance between the predicted values and ground truth of pose. Φ_e, Ψ_e and Ψ_d are trainable parameters of the sentence encoder, pose encoder and pose decoder respectively.

Combining equations 4 and 5 we get a joint translation loss,

$$\mathcal{L}_j = \mathcal{L}_c + \mathcal{L}_u \tag{6}$$

Jointly optimizing the loss $\mathcal{L}_j$ pushes z_x and z_y closer together improving generalizability and additionally trains the pose decoder which is useful for inference from the joint embedding space.

As $\mathcal{L}_j$ is a mutivariate function in $X_{1:N}$ and $Y_{1:T}$, coordinate descent [33] for optimizing the loss function is a natural choice and is described in Algorithm 1.

4.3. Training Curriculum

Cross modal pose forecasting can be a challenging task to train [5]. Starting with simpler examples before moving on to tougher ones can be beneficial to the training process [4, 37, 36].

The curriculum design commonly used for pose forecasting [5] is adapted for our joint model. We first optimize the model to predict 2 time steps conditioned on the complete sentence. This easy task helps the model learn very short pose sequences like leg motions for walking, hand motions for waving and torso motions for bending. Once the loss on the validation set starts increasing, we move on to the next stage in the curriculum. The model is now given twice the amount of poses for prediction. The complexity of the task is increased in every stage till the maximum time-steps (T) of prediction is reached. We describe the complete training process in Algorithm 1.

Algorithm 1 Learning language-pose joint embedding

procedure INITIALIZATION
  $\mathcal{X}_{train}, \mathcal{X}_{val}, \mathcal{Y}_{train}, \mathcal{Y}_{val} \leftarrow$ SplitData$(\mathcal{X}, \mathcal{Y})$
  MaxValLoss $\leftarrow$ inf
  $t \leftarrow 2$
procedure CURRICULUM
  while $t \leq T$ **do**
    for all $X_{1:N}, Y_{1:t} \in \mathcal{X}_{train}, \mathcal{Y}_{train}$ **do**
      $r \leftarrow$ CoinFlip() // For Coordinate Descent
      if $r == 0$ **then**
        $z \leftarrow p_e(X_{1:N}; \Phi_e)$ //Encoder
      else
        $z \leftarrow q_e(Y_{1:t}; \Psi_e)$ //Encoder
      $\hat{Y_{1:t}} \leftarrow q_d(z; \Psi_d)$ //Decoder
      loss $\leftarrow d(Y_{1:t}, \hat{Y_{1:t}})$
      $\Phi_e, \Psi_e, \Psi_d \leftarrow$ UpdateModelParams(loss)
    ValLoss $\leftarrow CalcValLoss(\mathcal{X}_{val}, \mathcal{Y}_{val})$
    if ValLoss $>$ MaxValLoss **then**
      $t \leftarrow 2t$
      MaxValLoss $\leftarrow$ inf

4.4. Optimization

For the distance metric $d(x, y)$ in Equation 4, 5 and 6, Smooth L1 loss (similar to Huber Loss [15]) is used which is defined as,

$$SmoothL1(x, y) = \begin{cases} 0.5(x-y)^2 & \text{for } |x-y| < 1 \\ |x-y| - 0.5 & \text{otherwise} \end{cases} \tag{7}$$

In contrast, Lin et. al.[17] uses L2 loss for $d(x, y)$. L2 loss is more sensitive to outliers than L1 loss due to its linearly proportional gradient with respect to the error, while L1 loss has a constant gradient of 1 or -1. But L1 Loss can become unstable when $|x - y| \approx 0$, due to oscillating gradients between 1 and -1. On the other hand, Smooth L1 is continuous and smooth near 0 and more generally for all $x, y \in \mathcal{R}$, hence it is more stable than L1 as a loss function.

5. Experiments

Joint language to pose modeling can be broken down into three core challenges,

1. **Prediction Accuracy by Joint Space**: How accurate is pose prediction from the joint embedding ?
2. **Human Judgment**: Which of the generated animation is more representative of the input sentence? Does the subjective evaluation correlate with the results from the objective evaluations?
3. **Modeling nuanced language concepts**: Is the model able to map nuanced concepts such as speed, direction and action in the generated animations?

Experiments are designed to evaluate these challenges of language grounded pose forecasting.

In the following subsections, the dataset and its preprocessing is briefly discussed which is followed by the evaluation metrics for both objective and subjective evaluations. Finally, design choices of the encoder and decoder models are described which are used to construct the baselines in the final subsection.

5.1. Dataset

Our models are trained an evaluated on KIT Motion-Language Dataset [25] which combines human motion with natural language descriptions. It consists of 3911 recordings (approximately 11.23 hours) which are re-targeted to a kinematic model of a human skeleton with 50 DoFs (6 DoF for the *Root* joint's orientation and position, while remaining 44 DoFs for arms legs, head and torso). The dataset also consists of 6278 English sentences (approximately 8 words per sentence) describing the recordings. This is more than the number of recordings as each recording has one or more descriptions which are annotated by human volunteers. We use 20% of the data as a randomly sampled held-out set for evaluating all models.

There is wide variety of motions in this dataset ranging from locomotion (e.g. walking, running, jogging), performing (e.g. playing violin/guitar), and gesticulation (e.g. waving). Many recording have adjectives to further describe the motion like speed (e.g. fast and slow), direction (left, right and forward), and number for periodic motions (e.g. walk 4 steps).

We use the pre-processing steps used in Holden et. al. [14]. All the frames of the motion are transformed such that body always faces the Z-axis. Joint rotation angles are transformed to 3D positions is the skeleton's local frame of reference with *Root* as the origin. *Root*'s position on XZ-plane and orientation along Y-axis is represented as velocity instead of absolute values.

Motion sequences are then sub-sampled to a frequency of 12.5 Hz down from 100Hz. This is low enough to bring enough variance between 2 time-steps for the decoder to train for a regression task, while not compromising on the human's perception of the animation [5].

5.2. Implementation Details

For pose encoder (q_e) a network of Gated Recurrent Units (GRUs) [9] is used in our model JL2P. The pose decoder (q_d) is the same except it has residual connection from the input to the output layer. This is similar to the pose decoder in Lin et. al. [17], except an extra layer to predict the trajectory (or Trajectory Predictor) is discarded in our model.

For langauge encoder (p_e), a network of Long-Short Term Memory Units (LSTMs) [12] is used. Each token of the sentence is converted into a distributional embedding using a pre-trained Word2Vec model [22]. [1]

5.3. Baselines

There has been limited work done in the domain of data-driven cross-modal translation from natural language descriptions to pose sequence generation. The closest work to our proposed approach is by **Lin et. al.** [17][2]. As mentioned in Section 2, their model does not follow a training curriculum and uses L2 loss as the loss function. Thier model also maps the language embeddings to an existing embedding space of poses instead of jointly learning it.

We also compare our model **JL2P** (see Section 4) with three ablations derived from itself. These ablations study the 3 main components of the model, joint embedding space, curriculum learning and Smooth L1 loss:

[1] We also train a variant of the model with BERT as the language encoder, but it did not show any significant improvements.

[2] As we could not find code or pre-trained models for this work, we use our own implementation and training on the same data as all other baselines

Models	Average Positional Error (APE) in mm										
	Mean	**Mean w/o Root**	**Root**	**Torso**	**Head**	**LArm**	**RArm**	**LHip**	**RHip**	**LFoot**	**RFoot**
Lin et. al.[17]	54.9***	50.0	151.6	26.6	35.4	61.3	61.6	32.2	32.1	63.3	63.2
JL2P w/o Curriculum	52.2***	47.9	139.2	24.2	32.5	57.3	57.2	30.6	30.7	62.9	63.2
JL2P w/o L1	51.7**	47.0	145.0	24.4	32.8	58.0	57.6	29.9	30.7	59.3	59.8
JL2P w/o Joint Emb.	50.4	45.7	143.3	24.0	**31.0**	55.6	**54.5**	29.7	29.5	**59.0**	59.5
JL2P	**49.5**	**45.4**	**131.1**	**23.0**	31.4	**55.3**	55.0	**28.6**	**29.0**	59.2	**58.8**

Table 1: Average positional error (APE) for JL2P , JL2P w/o Joint Emb., JL2P w/o L1, JL2P w/o Curriculum and Lin et. al.. Lower is better. Our models (JL2P and variants) show consistent increase in accuracy over Lin et. al. across all joints with the addition of components joint embedding, smooth L1 loss and curriculum learning. Two-tailed pairwise t-test between all models and JL2P where $***$- p<0.001, and $**$- p<0.01.

- **JL2P w/o Curriculum** - Training curriculum in Section 4.3 is dropped.
- **JL2P w/o L1** - L2 loss is used instead of Smooth L1 loss as the distance metric $d(x, y)$.
- **JL2P w/o Joint Emb.** - Instead of joint training as described in Section 4.2, autoencoder loss $\mathcal{L}_u$ minimized first followed by optimization of the cross-translation loss $\mathcal{L}_c$.

5.4. Objective Evaluation Metrics

All models are evaluated on the held-out set with a metric Average Position Error (APE). Given a particular joint j, it can be denoted as APE(j),

$$\text{APE}(p) = \frac{1}{\mathcal{Y}} \sum_{\mathcal{Y}} \|\hat{y}_t[j] - y_t[j]\|_2 \tag{8}$$

where $y_t[j]$ is the true location and $\hat{y}_t[j] \in \mathcal{Y}$ is the predicted location of joint j at time t

Another metric, Probability of Correct Keypoints (PCK) [3, 27], is also used as an evaluation metric.

5.5. User Study: Subjective Evaluation Metric

Joint language to pose generation is a subjective task, hence a human's subjective judgment on the quality of prediction is an important metric for this task.

To achieve this, we design a user study which asked human annotators to rank two videos generated by 2 different models but with same sentence as the input. One of the videos is generated by Lin et. al. and the other is either ground truth or generated by JL2P , JL2P w/o Curriculum, JL2P w/o Joint Emb., or JL2P L1. The annotators answer the following question for each pair of videos, *Which of the 2 generated animations is better described by "<sentence>"?*. To ensure that annotators spend enough time to decide, any annotations which took less than 20 seconds[3] were rejected. This study subjectively evaluates the preference of humans for generated animations by different models.

6. Results and Discussion

In this section we first use objective measures and then conduct a user study to get a subjective evaluation. Finally, we probe some qualitative examples to understand the effectiveness of the model in tackling the core challenges described in Section 5.

6.1. Prediction Accuracy by Joint Space

JL2P demonstrates at least a 9% improvement over Lin et. al. (see Table 1) for all joints. The maximum improvement around 15% is seen in the *Root* joint. Errors in *Root* prediction can lead to a "sliding-effect" of the feet; when the generation is translating faster than the frequency of the feet. Improvements in APE scores for long-term prediction, especially for *Root*, can help get rid of these artifacts in the generated animation.

When compared to its variants, JL2P loses maximum APE value when it is trained without curriculum (or JL2P w/o Curriculum). As discussed in Section 4.3, learning to

[3] each video is 8 seconds long at an average. We set a threshold of 20 seconds to give annotators 4 seconds to make their decision.

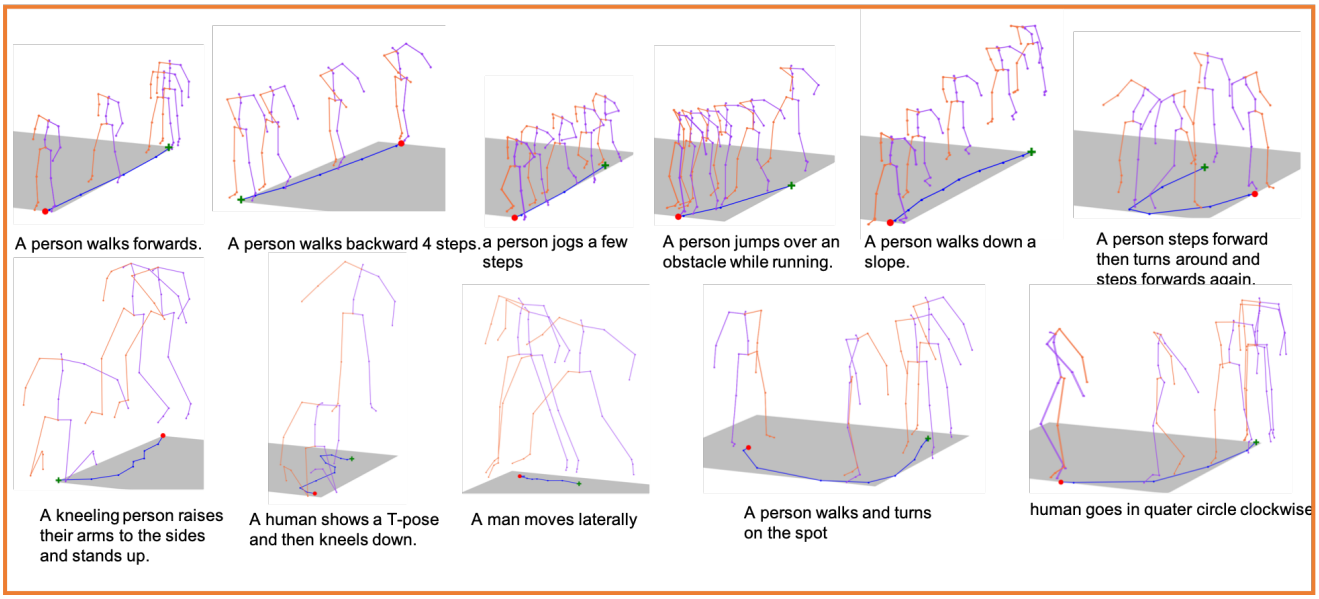

Figure 4: Renders of generated animations with a diverse set of sentences as input by our proposed model. Our model is able to change speed, direction and actions based on changes in the input sentence. Trajectory of the character is drawn with a blue line which starts at the green cross (x) and ends at the red circle (•).

predict shorter sequences before moving on to longer ones proves beneficial for pose generation. APE scores go down by 4%, if L2 loss is used instead of Smooth L1. In an output space as diverse as pose sequences, it becomes important to ignore outliers which may drive model to overfit. APE scores go down only by 1%, if the embedding space is not trained with the joint loss $\mathcal{L}_j$

APE values across time for JL2P of different parts of the body (*Root, Legs, Arms, Torso* and *Head*) are plotted in Figure 6. *Root*'s APE scores have the fastest rate of increase, followed by *Arms, Legs* and then *Head, Torso*. Two out of three coordinates of *Root* are represented as velocity which accumulates errors when integrated back to absolute positions; this is probably a contributing factor to the rapid increase of prediction error over time.

Our final objective metric is PCK. PCK values (for $35 \leq \sigma \leq 55$) on generated animations are compared among JL2P , its variants and Lin et. al. in Figure 7. JL2P and its ablations show a consistent improvement over Lin et. al. which further strengthen the claim about the prediction accuracy by our model's joint space.

6.2. Human Judgment

Human judgment is quantified by preference scores in Figure 5. Human preference of all our baseline models and ground truth are compared against Lin et. al. animations. JL2P has a preference of 75% which is shy of ground truth by 10%. Preference scores consistently drop with all the other variants of JL2P .

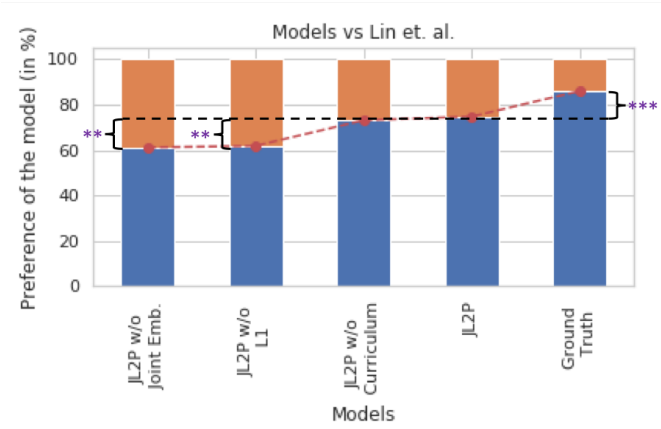

Figure 5: Preference scores of baseline models vs Lin et. al.. Blue bars denote the preference percentage of models marked on the horizontal axis. Our models (JL2P and variants) show consistent rise in preference over Lin et. al. with the addition of components joint embedding, smooth L1 loss and curriculum learning. **- $p < 0.01$ and ***- $p < 0.001$ for McNemar's test [21] on paired nominal data.

JL2P w/o Joint Emb. has the lowest preference score of 60% when ranked against Lin et. al. . It is still more preferred than Lin et. al. but far more unlikely to be picked when pitted against JL2P . This is an interesting change in trend, as removing joint loss from JL2P did not affect the objective scores significantly, but have lowered its hu-

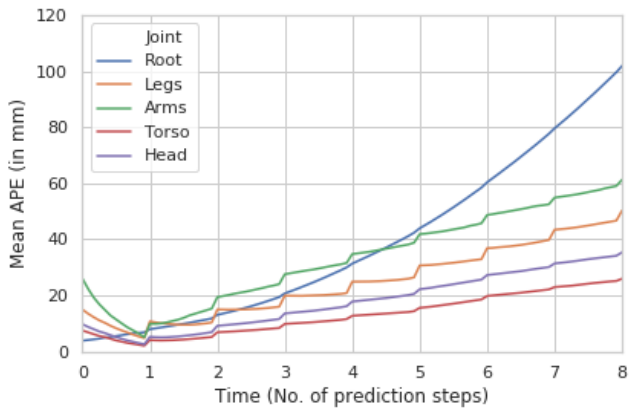

Figure 6: Plot of mean APE values across time for different parts of the body (*Root, Legs, Arms, Torso* and *head*) for JL2P. Lower is better. Generating trajectory of the animating character is harder than the other joints as *Root*'s APE blows up after around 500ms

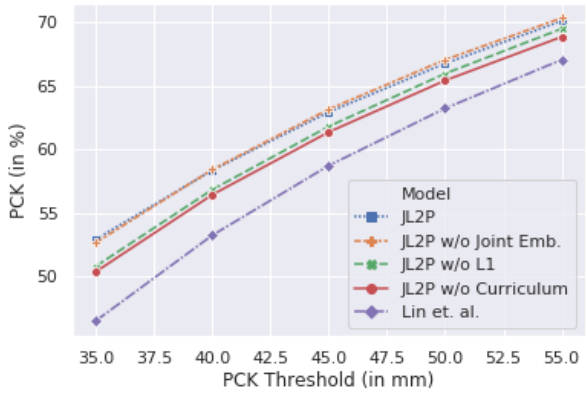

Figure 7: Plots of average Probability of Correct Keypoint (PCK) values over multiple values of thresholds (σ) for JL2P , JL2P w/o Joint Emb., JL2P w/o L1, JL2P w/o Curriculum and Lin et. al.. Our model JL2P shows consistent improvements over other baselines across a large range of thresholds. Higher values are better.

man preference by a significant fraction. This leads us to conclude that objective metrics are not enough to judge the performance of a model. Instead a combination of human judgment and objective metrics is necessary for evaluating pose generation models.

6.3. Modeling nuanced language concepts

Root joint decides the trajectory of the animation which is crucial for translating concepts like speed (e.g. fast, and slow), direction (e.g. left, right, forward and backward) from natural language to animation. We plot the trajectories generated by JL2P, ground truth and Lin et. al. for different sentences in Figure 3.

Modeling direction: Animations' trajectory for these sentences for JL2P is similar to that of the ground truth trajectories. In contrast, Lin et. al.'s trajectories tend to be semantically incorrect and have a slightly curved forward motion for these sentences.

Modeling speed: In the sentence, "A person runs very fast forward", JL2P is able to understand that the animation has to move faster. It is able to walk approximately the same distance as the ground truth in the same amount of time, hence has the same speed. In contrast, even though Lin et. al.'s motion is in the forward direction, it is not able to maintain the same speed as required by the sentence.

Modeling actions: In figure 4, we plot animations generated by a diverse set of sentences. JL2P is able to understand the action from the sentences, and is able to generate an animation corresponding to the action. JL2P is able to handle many actions ranging from *kneeling* (with complex leg motions) to *jogging* (with periodic hand and leg motion).

We show, via qualitative examples, that our model JL2P is able to model nuanced language concepts which are then reproduced in the animations generated at inference time.

7. Conclusions

In this paper, we proposed a neural architecture called Joint Language-to-Pose (or JL2P), which integrates language and pose to learn a joint embedding space in an end-to-end training paradigm. This embedding space can now be used to generate animations conditioned on an input description. We also proposed the use of curriculum learning approach which forces the model to generate shorter sequences before moving on to longer ones. We evaluated our proposed model on a parallel corpus of 3D pose data and human-annotated sentences with objective metrics to measure prediction accuracy, as well as with a user study to measure human judgment. Our results confirm that our approach, to learn a joint embedding in a curriculum learning paradigm by JL2P, was able to generate more accurate animations and are deemed more visually represented by humans than the state-of-the-art model.

8. Acknowledgements

This material is based upon work partially supported by the National Science Foundation (Award #1750439) and Oculus VR. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of National Science Foundation or Oculus VR. No official endorsement should be inferred.

References

[1] S. Agrawal and M. van de Panne. Task-based locomotion. *ACM Transactions on Graphics (TOG)*, 35(4):82, 2016.

[2] H. Ahn, T. Ha, Y. Choi, H. Yoo, and S. Oh. Text2action: Generative adversarial synthesis from language to action. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–5. IEEE, 2018.

[3] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *Proceedings of the IEEE Conference on computer Vision and Pattern Recognition*, pages 3686–3693, 2014.

[4] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48. ACM, 2009.

[5] Y.-W. Chao, J. Yang, B. Price, S. Cohen, and J. Deng. Forecasting human dynamics from static images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 548–556, 2017.

[6] C.-C. Chiu and S. Marsella. How to train your avatar: A data driven approach to gesture generation. In *International Workshop on Intelligent Virtual Agents*, pages 127–140. Springer, 2011.

[7] C.-C. Chiu, L.-P. Morency, and S. Marsella. Predicting co-verbal gestures: a deep and temporal modeling approach. In *International Conference on Intelligent Virtual Agents*, pages 152–166. Springer, 2015.

[8] H.-k. Chiu, E. Adeli, B. Wang, D.-A. Huang, and J. C. Niebles. Action-agnostic human pose forecasting. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1423–1432. IEEE, 2019.

[9] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

[10] P. Ghosh, J. Song, E. Aksan, and O. Hilliges. Learning human motion models for long-term predictions. In *2017 International Conference on 3D Vision (3DV)*, pages 458–466. IEEE, 2017.

[11] E. Hanser, P. Mc Kevitt, T. Lunney, and J. Condell. Scenemaker: Intelligent multimodal visualisation of natural language scripts. In *Irish Conference on Artificial Intelligence and Cognitive Science*, pages 144–153. Springer, 2009.

[12] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[13] D. Holden, T. Komura, and J. Saito. Phase-functioned neural networks for character control. *ACM Transactions on Graphics (TOG)*, 36(4):42, 2017.

[14] D. Holden, J. Saito, and T. Komura. A deep learning framework for character motion synthesis and editing. *ACM Transactions on Graphics (TOG)*, 35(4):138, 2016.

[15] P. J. Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics*, pages 492–518. Springer, 1992.

[16] Y. K. Hwang, P. C. Chen, and P. A. Watterberg. Interactive task planning through natural language. In *Proceedings of IEEE International Conference on Robotics and Automation*, volume 1, pages 24–29. IEEE, 1996.

[17] A. S. Lin, L. Wu, R. Corona, K. Tai, Q. Huang, and R. J. Mooney. 1. generating animated videos of human activities from natural language descriptions. *Learning*, 2018, 2018.

[18] X. Lin and M. R. Amer. Human motion modeling using dv-gans. *arXiv preprint arXiv:1804.10652*, 2018.

[19] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. Van Gool. Pose guided person image generation. In *Advances in Neural Information Processing Systems*, pages 406–416, 2017.

[20] M. Ma and P. Mc Kevitt. Virtual human animation in natural language visualisation. *Artificial Intelligence Review*, 25(1-2):37–53, 2006.

[21] Q. McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.

[22] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[23] Y. Pan, T. Mei, T. Yao, H. Li, and Y. Rui. Jointly modeling embedding and translation to bridge video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4594–4602, 2016.

[24] D. Pavllo, D. Grangier, and M. Auli. Quaternet: A quaternion-based recurrent model for human motion. *arXiv preprint arXiv:1805.06485*, 2018.

[25] M. Plappert, C. Mandery, and T. Asfour. The KIT motion-language dataset. *Big Data*.

[26] M. Plappert, C. Mandery, and T. Asfour. Learning a bidirectional mapping between human whole-body motion and natural language using deep recurrent neural networks. *Robotics and Autonomous Systems*, 109:13–26, 2018.

[27] T. Simon, H. Joo, I. A. Matthews, and Y. Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *CVPR*, volume 1, page 2, 2017.

[28] W. Takano and Y. Nakamura. Bigram-based natural language model and statistical motion symbol model for scalable language of humanoid robots. In *2012 IEEE International Conference on Robotics and Automation*, pages 1232–1237. IEEE, 2012.

[29] W. Takano and Y. Nakamura. Statistical mutual conversion between whole body motion primitives and linguistic sentences for human motions. *The International Journal of Robotics Research*, 34(10):1314–1328, 2015.

[30] K. Takeuchi, D. Hasegawa, S. Shirakawa, N. Kaneko, H. Sakuta, and K. Sumi. Speech-to-gesture generation: A challenge in deep learning approach with bi-directional lstm. In *Proceedings of the 5th International Conference on Human Agent Interaction*, pages 365–369. ACM, 2017.

[31] Y. Tang, L. Ma, W. Liu, and W. Zheng. Long-term human motion prediction by modeling motion context and enhancing motion dynamic. *arXiv preprint arXiv:1805.02513*, 2018.

[32] S. Taylor, T. Kim, Y. Yue, M. Mahler, J. Krahe, A. G. Rodriguez, J. Hodgins, and I. Matthews. A deep learning approach for generalized speech animation. *ACM Transactions on Graphics (TOG)*, 36(4):93, 2017.

[33] S. J. Wright. Coordinate descent algorithms. *Mathematical Programming*, 151(1):3–34, 2015.

[34] T. Yamada, H. Matsunaga, and T. Ogata. Paired recurrent autoencoders for bidirectional translation between robot actions and linguistic descriptions. *IEEE Robotics and Automation Letters*, 3(4):3441–3448, 2018.

[35] C. Yang, Z. Wang, X. Zhu, C. Huang, J. Shi, and D. Lin. Pose guided human video generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 201–216, 2018.

[36] J. Yang, S. E. Reed, M.-H. Yang, and H. Lee. Weakly-supervised disentangling with recurrent transformations for 3d view synthesis. In *Advances in Neural Information Processing Systems*, pages 1099–1107, 2015.

[37] W. Zaremba and I. Sutskever. Learning to execute. *arXiv preprint arXiv:1410.4615*, 2014.

[38] Y. Zhang, E. Tsipidi, S. Schriber, M. Kapadia, M. H. Gross, and A. Modi. Generating animations from screenplays. *CoRR*, abs/1904.05440, 2019.