8. Appendix

We provide architecture details in Sec. 8.1, results on long-term predictions in Sec. 8.2, PCK plots in Sec. 8.3, and more detailed ablation studies in Sec. 8.4.

8.1. Architecture Details

The RNN and Seq2seq models are implemented in Tensorflow [1]. For the QuaterNet-SPL model we extend the publicly available source code in Pytorch [23]. Our aim is to make a minimum amount of modifications to the baseline Seq2seq [20] and QuaterNet [25] models. In order to get the best performance on the new AMASS dataset, we fine-tune the hyper-parameters including batch size, learning rate, learning rate decay, cell type and number of cell units, dropout rate, hidden output layer size and teacher forcing ratio decay for QuaterNet.

Fig. 6 provides an overview over these models. The SP-layer replaces the standard dense layers, which normally use the context representation $\boldsymbol{h}_t$, i.e., GRU or LSTM state until time-step t, to make the pose vector prediction $\hat{\boldsymbol{x}}_t$. The SPL component follows the kinematic chain and uses the following network for every joint:

$$Linear(H) - ReLU - Linear(M) \ ,$$

where the hidden layer size per joint H is either 64 or 128 and the joint size M is 3, 4, or 9 for exponential map, quaternion, or rotation matrix pose representation, respectively (see Tab. 4). Similar to the H3.6M setup [14, 20] we use a 2-second seed $\boldsymbol{x}_{1:t-1}$ and 400-milisecond target sequences $\boldsymbol{x}_{t:T}$. The sequence $\boldsymbol{x}_{t:T}$ corresponds to the target predictions.

We train the baseline Seq2seq [20] and QuaterNet [25] models by using the training objectives as proposed in the original papers. The SPL variants, however, implement these objectives by using our proposed joint-wise loss. After an epoch of training we evaluate the model on the validation split and apply early stopping with respect to the joint angle metric. Please note that the early stopping metric is different than the training objective for all models.

	H3.6M			AMASS		
	SPL	Units	Cell	SPL	Units	Cell
RNN-SPL	sparse	64	GRU	dense	64	LSTM
Seq2seq-SPL	sparse	64	GRU	dense	64	LSTM
QuaterNet-SPL	sparse	128	GRU	sparse	128	GRU

Table 4: **SPL configuration.** *sparse* and *dense* refer to making a joint prediction by feeding only the immediate parent or all parent joints in the kinematic chain, respectively. Models use a hidden layer of either 64 or 128 units per joint. GRU cell outperforms LSTM on H3.6M while LSTM is consistently better on AMASS dataset. The vanilla models use their original setting with the reported cell.

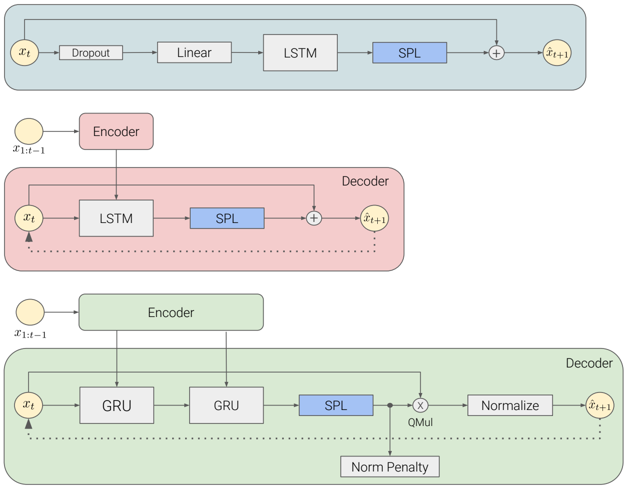

Figure 6: **Model overview**. *Top:* RNN-SPL *Middle:* Seq2seq-SPL, *Bottom:* Quaternet-SPL. Note that both Seq2seq and QuaterNet models follow sequence-to-sequence architecture where the encoder and decoder share the parameters. The 2-second seed sequence $\boldsymbol{x}_{1:t-1}$ is first fed to the encoder network to calculate the hidden cell state which is later used to initialize the prediction into the future. The dashed lines from the prediction to the input correspond to the sampling based training. In other words, the predictions are fed back during training.

RNN-SPL We use the rotation matrix pose representation with zero-mean unit-variance normalization, following teacher-forcing training. In other words, the model is trained by feeding the ground-truth pose $\boldsymbol{x}_t$ to predict $\hat{\boldsymbol{x}}_{t+1}$. The training objective is the proposed joint-wise loss with l_2-norm (see Sec. 3.3 in the paper) which is calculated over the entire seed $\boldsymbol{x}_{1:t-1}$ and target predictions $\hat{\boldsymbol{x}}_{t:T}$.

We do not follow a sampling-based training scheme. In the absence of such a training regularization, the model overfits to the likelihood (i.e., ground-truth input samples) and hence performs poorly in the auto-regressive test setup. We find that a small amount of dropout with a rate of 0.1 on the inputs makes the model robust against the exposure bias problem.

The dropout is followed by a linear layer with 256 units. We use a single LSTM cell with 1024 units. The vanilla RNN model makes the predictions by using

$$Linear(960) - ReLU - Linear(N) \ ,$$

where $N = K \cdot M$. We also experimented with GRU units instead of LSTM cells, but experimentally found that LSTMs consistently outperformed GRUs. Finally, we use the Adam

	Euler			Joint Angle			Positional			PCK (AUC)		
milliseconds	600	800	1000	600	800	1000	600	800	1000	600	800	1000
Zero-Velocity [20]	32.36	48.39	65.25	7.46	11.31	15.3	2.93	4.46	6.06	0.78	0.76	0.74
Seq2seq [20]*	41.96	71.63	109.45	8.75	15.57	24.43	3.13	5.55	8.76	0.76	0.71	0.66
Seq2seq-SPL	32.58	52.49	75.69	7.23	11.99	17.62	2.88	4.81	7.10	0.79	0.75	0.72
Seq2seq-sampling [20]*	27.72	42.19	58.01	5.96	9.21	12.79	2.34	3.64	5.07	0.81	0.79	0.77
Seq2seq-sampling-SPL	27.01	40.90	55.97	5.76	8.90	12.36	2.24	3.48	4.85	0.82	0.80	0.78
Seq2seq-dropout [20]*	31.20	50.62	73.09	6.59	10.93	15.98	2.53	4.18	6.09	0.80	0.76	0.73
Seq2seq-dropout-SPL	28.02	44.95	64.23	6.15	10.11	14.67	2.42	4.00	5.84	0.81	0.78	0.75
QuaterNet [25]*	27.08	41.32	56.66	5.88	9.21	12.84	2.32	3.64	5.09	0.82	0.79	0.77
QuaterNet-SPL	25.37	39.02	53.95	5.58	8.79	12.32	2.19	3.47	4.87	0.82	0.80	0.78
RNN	31.19	48.84	68.64	7.33	11.87	17.09	2.93	4.79	6.96	0.78	0.74	0.71
RNN-SPL	**24.44**	**38.02**	**53.06**	**5.04**	**8.08**	**11.50**	**1.94**	**3.14**	**4.49**	**0.84**	**0.81**	**0.79**

Table 5: **Long-term AMASS results** of the base models with and without the proposed structured prediction layer (SPL). For PCK we report the area-under-the-curve (AUC), which is upper-bounded by 1 (higher is better). Euler, joint angle and positional losses are lower-bounded by 0 (lower is better). "*" indicates our evaluation of the corresponding model on AMASS. "dropout" stands for dropout applied directly on the inputs. All models use residual connections. Note that models with our proposed SP-layer always perform better.

optimizer [15] with its default parameters. The learning rate is initialized with $1e^{-3}$ and exponentially decayed with a rate of 0.98 at every 1000 decay steps.

Seq2seq-SPL As proposed by Martinez *et al.* [20] we use the exponential map pose representation with zero-mean unit-variance normalization. The model consists of encoder and decoder components where the parameters are shared. The seed sequence $\boldsymbol{x}_{1:t-1}$ is first fed to the encoder network to calculate the hidden cell state which is later used by the decoder to initialize the prediction into the future (i.e., $\hat{\boldsymbol{x}}_{t:T}$). Similarly, the training objective is calculated between the ground-truth targets $\boldsymbol{x}_{t:T}$ and the predictions $\hat{\boldsymbol{x}}_{t:T}$. We use the proposed joint-wise loss with l_2-norm.

In our AMASS experiments, we find that a single LSTM cell with 1024 units performs better than a single GRU cell. In the training of the Seq2seq-sampling model, the decoder prediction is fed back to the model [20]. The other two variants, Seq2seq-dropout (with a dropout rate of 0.1) and Seq2seq (see Tab. 2 in the paper), are trained with ground-truth inputs similar to the RNN models. Similarly, the vanilla Seq2seq model has a hidden output layer of size 960 on AMASS dataset.

We use the Adam optimizer with its default parameters. The learning rate is initialized with $1e^{-3}$ and exponentially decayed with a rate of 0.95 at every 1000 decay steps.

QuaterNet-SPL We use the quaternion pose representation without any further normalization on the data [25]. The data is pre-processed following Pavllo *et al.*'s suggestions to avoid mixing antipodal representations within a given sequence. QuaterNet also follows the sequence-to-sequence architecture where the seed sequence is used to initialize the cell states. As in the vanilla model, the training objective is based on the Euler angle pose representation. More specifically, the predictions in quaternion representation are converted to Euler angles to calculate the training objective.

The model consists of two stacked GRU cells with 1000 units each. In contrast to the RNN and Seq2seq models, the residual velocity is implemented by using quaternion multiplication. Moreover, the QuaterNet model applies a normalization penalty and explicitly normalizes the predictions in order to enforce valid rotations. As proposed by Pavllo *et al.* [25], we exponentially decay the teacher-forcing ratio with a rate of 0.98. The teacher-forcing ratio determines the probability of using ground-truth poses during training. Over time this value gets closer to 0 and hence increases the probability of using the model predictions rather than the ground-truth poses. Similar to the vanilla RNN and Seq2seq models, a hidden output layer of size 960 performed better on AMASS dataset.

Finally, the model is trained by using the Adam optimizer with its default parameters. The learning rate is initialized with $1e^{-3}$ and exponentially decayed with a rate of 0.96 after every training epoch.

8.2. Long-term Prediction on AMASS

In Tab. 5, we report longer-term prediction results as an extension to the results provided in Tab. 2 in the main paper. Please note that all models are trained to predict 400-ms. In fact, the Seq2seq and QuaterNet models have been proposed to solve short-term prediction tasks only.

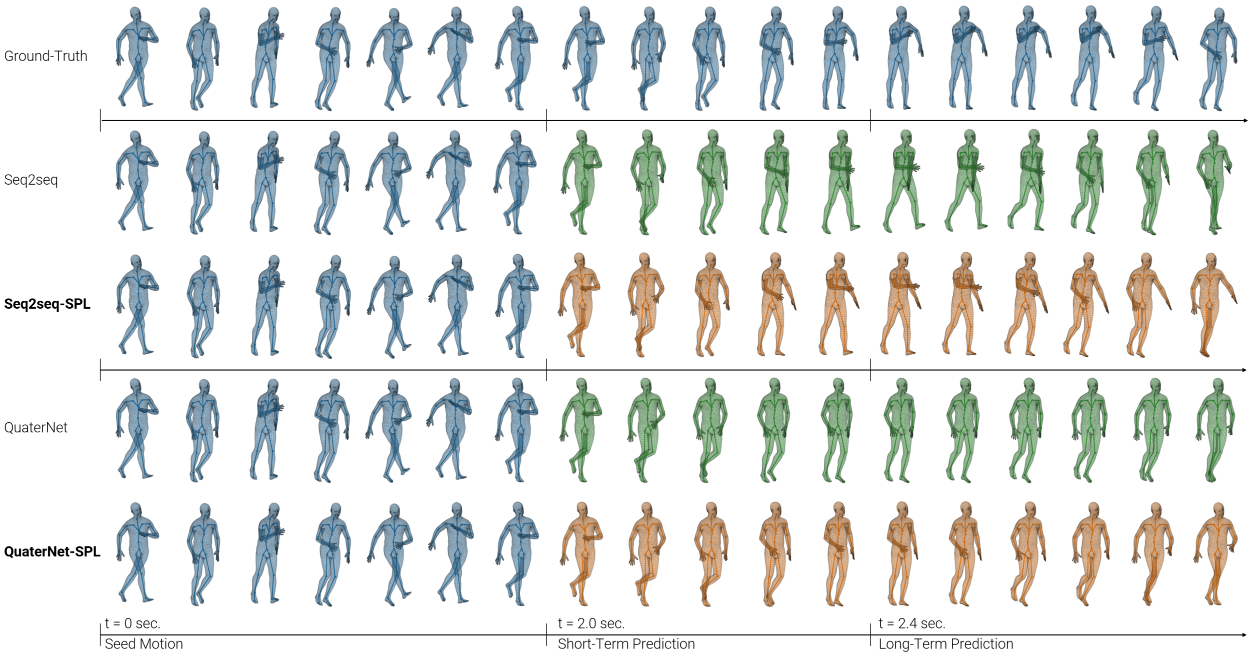

Figure 7: **Qualitative Comparison on AMASS.** We use a 2-second seed sequence and predict the next 1 second (60 frames). The last pose of the seed and the first pose of the prediction sequences are consecutive frames. In green (2nd and 4th row) are results from the vanilla versions of Seq2seq and QuaterNet, respectively. In orange (3rd and 5th row) are results when augmenting the vanilla model with our SP-layer. Although the SPL-variants shown here are still outperformed by the RNN-SPL shown in the main paper, they still show slight improvement over their non-SPL counterparts.

	Walking				Eating				Smoking				Discussion			
milliseconds	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400
RNN-mean	0.319	0.515	0.771	0.900	0.242	0.384	0.583	0.742	0.264	0.493	0.984	0.967	0.312	0.668	0.945	1.040
RNN-per-joint	0.324	0.534	0.816	0.950	0.233	0.391	0.616	0.776	0.258	0.483	0.961	0.932	0.312	0.675	0.969	1.067
RNN-SPL-indep.	0.288	0.453	0.720	0.836	0.228	0.366	0.575	0.736	0.258	0.482	0.947	0.916	0.313	0.676	0.962	1.064
RNN-SPL-random	0.298	0.473	0.758	0.863	0.227	0.354	0.578	0.717	0.263	0.490	0.956	0.925	0.311	0.677	0.975	1.079
RNN-SPL-reverse	0.302	0.483	0.725	0.849	0.225	0.344	0.557	0.721	0.264	0.494	0.96	0.929	0.312	0.679	0.960	1.050
RNN-SPL	0.264	0.413	0.669	0.772	0.205	0.326	0.559	0.721	0.260	0.486	0.958	0.930	0.307	0.667	0.950	1.049

Table 6: **H3.6M ablation study.** Comparison of SPL with different joint configurations and the proposed per-joint loss on H3.6M. Each model entry corresponds to an average of several runs with different initialization.

Consistent with the short-term prediction results shown in the main paper, our proposed SP-layer always improves the underlying model performance. While QuaterNet-SPL is competitive, RNN-SPL yields the best performance under different metrics.

In Fig. 7 we show more qualitative results for QuaterNet and Seq2seq when augmented with our SP-layer. Please refer to the supplemental video for more qualitative results.

8.3. PCK Plots

We provide additional PCK plots for 100, 200, 300 and 400 ms prediction horizon in Fig. 8. Please note that shorter time horizons do not use the entire range of thresholds ρ to avoid a saturation effect.

8.4. Ablation Study

The full ablation study on H3.6M and AMASS is shown in Tab. 6 and 7, respectively. For an explanation of each entry, please refer to the main text in Sec. 6.3.

	Euler				Joint Angle				Positional				PCK (AUC)			
milliseconds	100	200	300	400	100	200	300	400	100	200	300	400	100	200	300	400
RNN-mean	1.65	5.21	10.24	16.44	0.318	1.057	2.157	3.570	0.122	0.408	0.838	1.396	0.886	0.854	0.861	0.832
RNN-per-joint	1.33	4.15	8.16	13.13	0.230	0.758	1.550	2.573	0.086	0.287	0.590	0.986	0.923	0.897	0.901	0.877
RNN-SPL-indep.	1.30	4.08	8.04	12.96	0.228	0.750	1.537	2.552	0.085	0.283	0.587	0.982	0.924	0.897	0.901	0.878
RNN-SPL-random	1.31	4.09	8.03	12.98	0.228	0.749	1.533	2.547	0.086	0.284	0.586	0.980	0.924	0.897	0.901	0.878
RNN-SPL-reverse	1.31	4.10	8.08	13.03	0.229	0.749	1.532	2.543	0.086	0.282	0.582	0.973	0.924	0.897	0.902	0.878
RNN-SPL	1.29	4.04	7.95	12.85	0.227	0.744	1.525	2.533	0.085	0.282	0.582	0.975	0.924	0.898	0.902	0.878

Table 7: **AMASS ablation study.** Comparison of SPL with different joint configurations and the proposed per-joint loss on AMASS. Each model entry corresponds to an average of several runs with different initialization.

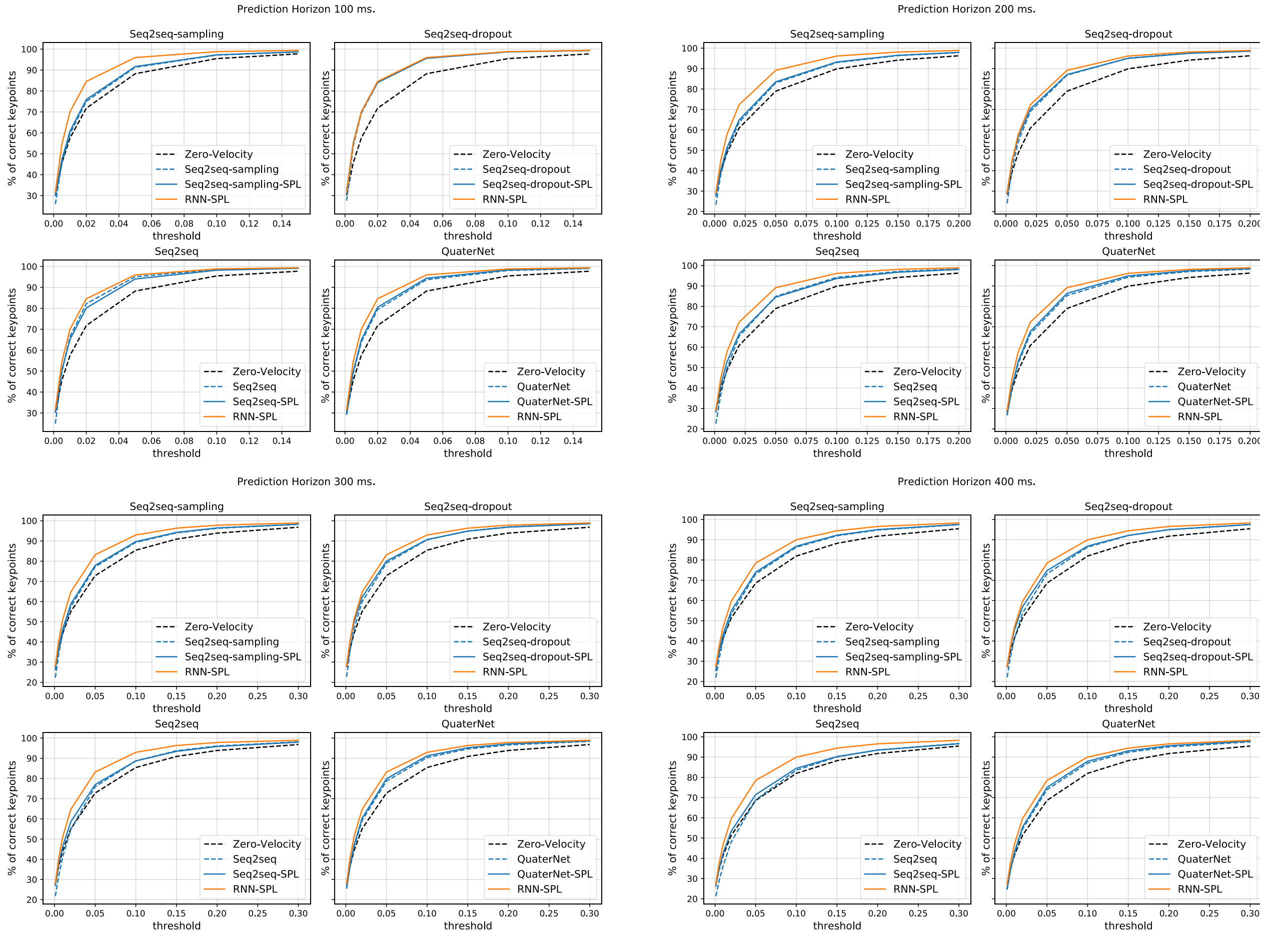

Figure 8: **PCK Curves** of models with and without our SP-layer (dashed lines) on AMASS for 100, 200, 300, and 400 milliseconds (*top left* to *bottom right*).