Context-aware Human Motion Prediction

Enric Corona Albert Pumarola Guillem Alenyà Francesc Moreno-Noguer
Institut de Robòtica i Informàtica Industrial, CSIC-UPC, 08028, Barcelona, Spain {ecorona, apumarola, galenya, fmoreno}@iri.upc.edu

Abstract

The problem of predicting human motion given a sequence of past observations is at the core of many applications in robotics and computer vision. Current state-of-the-art formulate this problem as a sequence-to-sequence task, in which a historical of 3D skeletons feeds a Recurrent Neural Network (RNN) that predicts future movements, typically in the order of 1 to 2 seconds. However, one aspect that has been obviated so far, is the fact that human motion is inherently driven by interactions with objects and/or other humans in the environment.

In this paper, we explore this scenario using a novel context-aware motion prediction architecture. We use a semantic-graph model where the nodes parameterize the human and objects in the scene and the edges their mutual interactions. These interactions are iteratively learned through a graph attention layer, fed with the past observations, which now include both object and human body motions. Once this semantic graph is learned, we inject it to a standard RNN to predict future movements of the human/s and object/s. We consider two variants of our architecture, either freezing the contextual interactions in the future of updating them. A thorough evaluation in the “Whole-Body Human Motion Database” [29] shows that in both cases, our context-aware networks clearly outperform baselines in which the context information is not considered.

1. Introduction

The ability to predict and anticipate future human motion based on past observations is essential for interacting with other people and the world around us. While this seems a trivial task for a person, it involves multiple sensory modalities and complex semantic understanding of the environment and the relations between all objects in it. Modeling and transferring this kind of knowledge to autonomous agents would have a major impact in many different fields, mainly in human-robot interaction [30] and autonomous driving [47], but also in motion generation for computer graphics animation [31] or image understanding [10].

The explosion of deep learning, combined with large-scale datasets of human motion such as Human3.6M [24] or the CMU motion capture dataset [34], has led to a significant amount of recent literature that tackles the problem of forecasting 3D human motion from past observations [14, 25, 43, 20, 3, 37, 15, 42, 26, 66]. These algorithms typically formulate the problem as sequence-to-sequence task, in which past observations represented 3D skeleton data are injected to a Recurrent Neural Network (RNN) which then predicts movements in the near future (less than 2 seconds).

Nevertheless, while promising results have been achieved, we argue that the standard definition of the problem used so far lacks an important factor, which is the influence of the rest of the environment on the movement of the person. For instance, if a person is carrying a box, the configuration of the body arms and legs will be highly constrained by the 3D position of that box. Discovering such interrelations between the person and the object/s of the context (or another person he/she is interacting with), and how these interrelations constrain the body motion, is the principal motivation of this paper.

In order to explore this new paradigm, we devise a context-aware motion prediction architecture, that models the interactions between all objects of the scene and the human using a directed semantic graph. The nodes of this graph represent the state of the person and objects (e.g. positional features) and the edges their mutual interactions. These interactions are iteratively learned with the past observations of the human and objects motion and fed into a standard RNN which is then responsible for predicting the future movement of all elements in the scene (for both rigid objects and non-rigid human skeletons). Additionally, we propose a variant of this model that also predicts the evolution of the adjacency matrix representing the interaction between the elements of the scene.

Presumably, one of the reasons why current state-of-the-art has not considered an scenario like ours is because all methods are trained and evaluated on benchmarks (mostly
the aforementioned Human3.6M dataset [24]) annotated only with human motion. In this paper, we thoroughly evaluate our approach in the “Whole-Body Human Motion Database” [29], that contains about 200 videos of people performing several tasks and interacting with objects. This dataset is annotated with MoCap data for the humans and rigid displacement for the rest of objects, being thus, a perfect benchmark to validate our ideas. We also evaluate our method in the CMU MoCap database [34] with only two people being tracked. The results obtained in both datasets show that our methodology is able to accurately predict the future motion of people and objects while simultaneously learning very coherent interaction relations. Additionally, all context-aware versions, clearly outperform the baselines which uniquely rely on human past observations of the human (see Fig. 1). Since all previous works evaluate their methods using past observations of ground truth skeletons, we finally discuss the applicability of state-of-art motion prediction methods, with an ablation study of our models and baselines when considering noisy observations.

2. Related work

Human motion prediction. Since the release of large-scale MoCap datasets [24, 29], there has been a growing interest in the problem of estimating 3D human pose from single images [5, 24, 49, 56, 55, 44, 57]. More recently, the community is focusing in predicting 3D human motion from past observations. Most approaches build upon RNNs [14, 43, 20, 3, 50, 1] that encode historical motion of the human and predict the future configuration that minimizes different sort of losses. Martinez et al. [43], for instance, minimize the L2 distance and provide one of the baselines in our work. This work also compares against a zero-velocity baseline, which despite steadily predicting the last observed frame, yields very reasonable results under the L2 metric. This phenomenon has been recently discussed by Ruiz et al. [54], that argue that L2 distance is not an appropriate metric to capture the actual distribution of human motion, and that a network trained using only this metric is prone to converge to a mean body pose. To better capture real distributions of human movement, recent approaches use adversarial networks [17, 2] in combination with geometric losses [3, 20, 54, 33].

There exist alternative approaches other than RNNs. For instance, Jain et al. [25] consider a hand-crafted spatial-temporal graph adapted to the skeleton shape. Li et al. [37] use Convolutional Neural Networks to encode and decode skeleton sequences instead of RNNs.

All methods described in this section formulate the human-prediction problem without considering the context information. In this paper, we aim to fill this gap.

Rigid 3D object motion prediction. While there is a vast amount of works on 3D object reconstruction [51, 19, 41], detection [9, 18, 11] and tracking [8, 4], only very few approaches address the problem of predicting future rigid motion [6, 63, 32, 59]. Among these, it is worth to mention Byravan et al. [6], that predict the future 3D pose given an image of an object and the action being applied to it. In our case, the action applied to each object is implicitly encoded in the previous observations.

Human-Object Interaction (HOI). Even though our work does not aim to identify Human-Object relationships, we have been inspired by a few papers on this topic. The standard formulation of the problem consists in representing an image with several detected objects and people as a graph encoding the context [22, 45, 53, 39, 16], or some other structured representation [36, 65, 12]. The most recent approaches [37, 53, 22] extract features of the detected entities using some image-based classification CNNs. Then, they compare pairs of features to predict their mutual interaction. Qi et al. [53] refine the representations and predicted interactions in a recursive manner. In this work, we use a similar idea to progressively refine the estimation of the interactions between objects.

Graph-based context reasoning. A few works leverage context information to boost the performance of different tasks [48, 38, 23, 35, 46]. Graph Convolutional Networks
(GCNs) [28] were recently proposed for improved semi-supervised classification. Jain et al. [25] used Structural RNNs to model spatio-temporal graphs. Wang et al. [61] propose to use GCNs, in which the interactions between objects depend on the intersection over union of their detected bounding boxes. Chen et al. [10] introduce an approach for image segmentation in which features from a 2D image coordinate space are represented in a graph reasoning space.

3. Problem formulation

Recent methods for human motion prediction consist of a model $M$, typically a deep neural network, that encodes motion from time $t_0$ until $t - 1$. The goal is then to predict future human motion until $t_f$, namely $P_{t:t_f}$, where $P$ stands for the human pose represented by 3D joint coordinates. Previous approaches have formulated the problem as $M : P_{t:t-1} \rightarrow P_{t:t_f}$, i.e. future motion is estimated only from past observations. In this paper, we conjecture that future motion is also driven by the context and the action the human is performing. We therefore consider other objects $O$ of type $T$ in the scene with which the human may interact. The objects can be other people or any object in the scene. We will design our approach to be able to predict the motion of such objects of the context.

Additionally, the influence that objects will have in the future motion of other objects is unclear. Thus, we also aim to build a model that learns these interactions in an unsupervised manner. Considering all this, we reformulate our problem as the estimation of the following mapping:

$$M : \{P_{t:t-1}, O_{t:t-1}, T\} \rightarrow \{P_{t:t_f}, O_{t:t_f}, I_{t:t_f}\},$$

where $I$ corresponds to the predicted interactions.

4. Approach

Figure 2 shows the main architecture used in this work. It consists of two branches that separately process human motion and object relationships. We use the latter to obtain a representation for all the observed entities, including the human, which we then use to predict both human and object motion prediction. We next describe these two branches.

4.1. Human motion branch

This branch builds upon the RNN network proposed by Martínez et al. [43]. This model, depicted in blue in Figure 2, is based on a residual architecture [21] that, at each step, uses a fully connected layer to predict the velocity of the body joints. As in a typical sequence-to-sequence network, the predictions are fed to the next step.

4.2. Context branch

The context information is represented using a directed graph structure where each node denotes an object or person. We then store a state for each entity and frame, encoding context information relevant to each node. These states are iteratively refined as new observations are processed.

Object representation. At each frame $t$, we define a matrix $X_t \in \mathbb{R}^{N \times F_t} = [O_t, T_t, P_t]$ that gathers the representation of all $N$ nodes. $F_0$ is the length of the state vector of each node. This state vector contains the object 3D bounding box $O_t$, their object type $T$ as a one-hot vector, and the joints of the person $P_t$. If the node does not correspond to a person, the joints in the representation are set to a zero vector of same size. The object type helps to identify the task the human is performing and the motion defined for that task.
By doing this, we aim to capture the semantic difference between the motion of a person when handling a knife or when using a whisk.

**Modelling contextual object representations.** Recent works on Graph Convolutional Networks (GCNs) [28] have shown very promising results in a variety of problems requiring the manipulation of graph-structured data. In GCNs, a feature vector of a certain node \( R_i \) is expressed as a function of other nodes \( x \), as \( R_i = \sigma(\sum_j A_{ij} W x_j) \), where \( W \) are trainable weights, \( \sigma \) is an activation and \( N \) the number of nodes of the graph connected to the \( i \)-th node. \( \tilde{A} \in \mathbb{R}^{N \times N} \) is a normalized weighted adjacency matrix that defines interactions between nodes.

Graph Attention Networks (GATs) [58] have been proposed as an extension of GCNs, and introduce an attention model on every graph node. In this paper we also investigate the use of Edge Convolutions [62], which are indeed very similar to GATs. In ECs the update rule for a feature vector of each entity considers the representations of other relevant objects as follows:

\[
R_i = \sigma(\sum_j \tilde{A}_{ij} W \left[ x_i; x_i - x_j \right]). \tag{2}
\]

The intuition behind this equation is that \( x_i \) encodes a global representation of the node, while \( x_i - x_j \) provides local information. EC proposes combining both types of information in an asymmetric graph function.

We keep track of the context representations during all observations through a second RNN. Each node on the scene has a hidden state \( H \) that is updated every frame \( t \):

\[
H_t^{i+1} = \text{RNN}(R_t^i, H_t^i). \tag{3}
\]

**Learning interactions.** As we shall see in the experimental section, we initially evaluate a simplified version of our Context-RNN (C-RNN) that uses a heuristic to define the adjacency matrices, setting \( A_{ij} = 1 \) if the center of gravity of objects \( i \) and \( j \) is closer than 1 meter.

In practice, interactions between entities are not known a priori, and furthermore, they change over time. Our goal is to automatically learn these changing interactions with no supervision. For this purpose we devise an iterative process in which, for the first frame, we set \( A_0 \) to a diagonal matrix, i.e. \( A_0 = I_N \), meaning that the initial hidden representation of every object depends only on itself. We then predict the value of the interaction between two objects given the hidden state of both. We consider asymmetric weighted adjacency matrices, that for a frame \( t \) are estimated as:

\[
A_{ij}^t = g(H_t^i, H_t^j - H_t^i), \tag{4}
\]

with similar structure as in Eq. 2. The function \( g(\cdot) \) represents the output of a neural network layer, in our case a fully connected. We normalize the interactions for each node using a Softmax function, which we shall denote \( \tilde{A} \).

Intuitively, we can consider this as a complete graph, where a graph attention mechanism [58] decides on the strength of interactions based on past observations. Note that while existing works typically use binary adjacency matrices from ground truth relationships [28], spatial assumptions [61] or K-NN on node representations [62], in this work we consider a differentiable continuous space of interactions, learned using back-propagation. In the rest of the paper we will denote the models that learn interactions with the suffix “-LI” (e.g. C-RNN+LI).

**Object motion prediction.** We propose two methods that exploit context at different levels. First, in the blue-brown modules of Fig. 2, we consider a model that reasons about the past context observations and iteratively improves hidden representations. The refined context representation of the human node is concatenated to the baseline branch (in blue) representation at every time step, and used by a fully connected layer to predict human velocity in that step. This is followed by a residual layer that yields skeleton poses.

Our second approach consists of the complete model depicted in Fig. 2 which, apart from past context, predicts object motion for all objects using a residual fully connected layer on each object hidden state. Analogous to the human motion branch, the predicted positions are forwarded to the next step, allowing to extend the context analysis into the future. The joints in the feature representations for those nodes describing people are also updated with the joint predictions of the human branch.

Additionally, when tracking several people, the human motion branch is repeated for each of them, and the model provides complete future motion for all available entities.

In the rest of the document, we will denote the models that predict object motion with the suffix “-OPM”.

**5. Implementation details**

Our model builds on the residual architecture of Martínez et al. [43] to allow an unbiased comparison with their work. The size of the human and object RNN hidden representations are 1024 and 256, respectively.

After the motion seed, we sample an observation every 100 ms. In all experiments, we encode and decode 10 (1 sec.) and 20 frames (2 sec.) respectively. Larger encoding times did not help in improving the results and significantly increased training time. We augment the train set through random rotation over the height \( Z \) in the range \((-180, 180)^\circ\) and random translation \( X, Y \in (-1500, 1500)\) mm.

We use a similar approach as in [53] to obtain the adjacency matrix. We build a 4D matrix \( A \) such that \( A_{ij} \) contains the hidden representations \([H_i; H_j - H_i] \) of nodes \( i \) and \( j \), extending over the channel dimension. The function \( g(\cdot) \) is formed by two Convolutional Layers of output kernel size 1 to make computation faster. We do not use bias term in these Convolutional layers nor in the Edge Convolutions.
table 1: class-specific models results. In this table, every action is independently trained. The results report the mean Euclidean error (in mm), for the 2s prediction of the human motion (top) and object motion (bottom). In all cases, 1s of past observations is provided. The context-based models we propose in this paper are those with the suffixes “OMP” and “LI”. They provide the best results in most sequences.

<table>
<thead>
<tr>
<th>Human motion</th>
<th>Passing objects</th>
<th>Grasping objects</th>
<th>Cutting food</th>
<th>Mixing objects</th>
<th>Cooking</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time (s)</td>
<td>0.5 1 1.5 2</td>
<td>0.5 1 1.5 2</td>
<td>0.5 1 1.5 2</td>
<td>0.5 1 1.5 2</td>
<td>0.5 1</td>
</tr>
<tr>
<td>ZV [43]</td>
<td>34 81 120 153</td>
<td>54 132 198 258</td>
<td>54 132 198 258</td>
<td>24 53</td>
<td>70 80</td>
</tr>
<tr>
<td>RNN [43]</td>
<td>50 99 132 162</td>
<td>82 158 211 254</td>
<td>54 132 198 258</td>
<td>24 53</td>
<td>70 80</td>
</tr>
<tr>
<td>QuarterNet</td>
<td>62 145 211 267</td>
<td>208 209 238 292</td>
<td>87 211 308 399</td>
<td>192 345 39</td>
<td>87 121 144</td>
</tr>
<tr>
<td>C-RNN</td>
<td>47 102 141 177</td>
<td>76 149 203 247</td>
<td>49 100 124 158</td>
<td>70 158 214 247</td>
<td>26 53 63 69</td>
</tr>
<tr>
<td>C-RNN+OMP</td>
<td>53 99 127 155</td>
<td>128 154 197 239</td>
<td>49 96 121 149</td>
<td>61 127 168 199</td>
<td>29 55 65 70</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Object motion</th>
<th>Passing objects</th>
<th>Grasping objects</th>
<th>Cutting food</th>
<th>Mixing objects</th>
<th>Cooking</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time (s)</td>
<td>0.5 1 1.5 2</td>
<td>0.5 1 1.5 2</td>
<td>0.5 1 1.5 2</td>
<td>0.5 1 1.5 2</td>
<td>0.5 1</td>
</tr>
<tr>
<td>ZV</td>
<td>48 118 181 237</td>
<td>65 152 226 289</td>
<td>29 70 104 132</td>
<td>50 126 188 229</td>
<td>16 33 44 53</td>
</tr>
<tr>
<td>RNN</td>
<td>49 107 154 198</td>
<td>64 139 201 257</td>
<td>29 70 105 134</td>
<td>47 113 166 199</td>
<td>17 36 48 58</td>
</tr>
<tr>
<td>C-RNN+OMP</td>
<td>44 92 122 150</td>
<td>55 103 136 167</td>
<td>31 64 83 97</td>
<td>29 65 90 110</td>
<td>15 33 46 56</td>
</tr>
<tr>
<td>C-RNN+OMP+LI</td>
<td>44 91 119 142</td>
<td>58 112 152 186</td>
<td>29 62 81 92</td>
<td>51 106 145 171</td>
<td>16 34 46 55</td>
</tr>
</tbody>
</table>

Table 2: Training with all actions simultaneously. For each method we train a single model using all actions simultaneously. See also caption in Table 1.
Figure 3: **Qualitative motion generation up to two seconds.** Left: Predicted sample frames of our approaches and the baselines. Center: Detail of the predictions obtained with our approaches, compared with the ground truth. Human and object motion are represented from light blue to dark blue and light green to dark green, respectively. Actions, from top to bottom are: A human supports on a table to kick a box, human leaning on a table, and two people (one of them standing on a ladder) passing an object. Right: Predicted adjacency matrices representing the interactions learned by our model. Note that these relations are directional (e.g. in the last example the ladder highly influences the motion of the Human#1 (50%) but the human has little influence over the ladder (11%). Best viewed in color with zoom.

RNN. We also consider a Zero-Velocity (ZV) baseline that constantly predicts the last observed frame. We also compare to QuaterNet [50] using their available code, to predict absolute motion prediction. For object motion prediction, we also use a ZV and RNN models [43], where the position of an object is defined by its 3D bounding box.

**Our models.** We run our context-aware models (C-RNN), incrementally adding the main ideas described in the paper. The basic C-RNN in our experiments uses the spatial heuristic described in Section 4.2 where interactions depend only on the distance between objects. This model processes context during the past frames, and then uses the last hidden state of the human node for human motion prediction at each step. This is extended by additionally predicting object motion (OMP) and recomputing object interaction from the previous assumption on the predicted positions. We then evaluate the efficiency of our model for learning interactions (LI). Like in the previously defined experiments, we evaluate a model that considers past contextual information and a model that prolongs object analysis into the future.

**Evaluation metric.** Previous works on human motion prediction focus mainly on predicting relative motion [43, 20, 50], using joint angles. However, our model reasons about the full scene and is able to predict absolute motion in Cartesian coordinates. Therefore, we use the mean Euclidean Distance (in mm) between predictions and real future motion, obtained from the unnormalized predictions in the 3D space. For human motion prediction, we take into account the 18 joints defined in the human skeleton. For objects, we consider the eight 3D vertices of their bounding boxes.
Figure 4: **Average interactions refined by the model during the past observations of the context.** In the left and center plots, we depict relevant interactions for *table cleaning* and *moving box* activities respectively. In the first case, notice the table affects significantly the sponge and human, which initially moves towards the table to clean it. Similarly, in the second case, the human moves towards a box on the ground, picks it up and puts it on the table. The right plot shows average self-interaction percentages among all the test samples, for relevant object types. We found that non-moving objects like tables or ladders consistently have very little influence from other objects. Likewise, passive objects that are often moved by a human, such as knives or bottles, are more influenced by them and leave self-influence relatively low.

6.2. Results on the WBHM Dataset

**Quantitative results.** Table 1 summarizes the performance of class-specific models trained on different activities. Table 2 provides results at much higher temporal resolution for models trained using all the dataset, reporting the mean Euclidean distance between predictions and ground truth every 100 ms. In all cases, 1 second of past observations is provided and 2 seconds are predicted.

The performance of models that consider a threshold-based binary interaction vary significantly between classes, suggesting they are effectively unable to understand the context as done by models that learn the actual interactions (*LI*). Notice that even the basic *C-RNN* does not yet provide a consistent improvement compared to state-of-art models. The same model that additionally learns interactions (*C-RNN+LI*) obtains a significant boost in most cases. Nonetheless, activities such as passing objects or grasping require attending to items that are at variable distances.

Regarding the complexity of the scene, most improvement comes from scenes with a small number of objects where interactions are well defined and actions are more predictable. For cooking activities, there are several objects in a table next to the human. Different motion options are possible and, as uncertainty grows, the model seems unable to confidently understand interactions. Because of this, context-aware models do not provide such a significant improvement as in previous activities. Considering all actions simultaneously seems to favor even more the context-aware approaches and, specially, those that learn interactions (*C-RNN+LI* and *C-RNN+OPM+LI*).

**Qualitative results.** Figure 3-left shows the motion generation results of our two main models, compared to the baseline [43] on different classes. We did not include the Zero-Velocity baseline as it does not provide interesting motion even though it has remained a difficult baseline on uncertain activities. We have marked some specific frames in which context-aware approaches improve the RNN baseline.

For human motion prediction, poses generated are frequently more semantically-related to their closest objects than context-less models. For instance, as shown in the last action of Figure 3, people holding objects tend to move the relevant hand. For object motion prediction, context-less model predictions hardly move from their original position.

Regarding the interactions predicted by the model, we notice coherent patterns in many activities. For example, drinking videos generate strong Cup-Human relationships. In Figure 4, we represent the average predicted interactions for different actions. These are gathered from the *C-
We show this in Figure 5. In this work, we explore a context-aware motion prediction architecture, using a semantic-graph representation where objects and humans are represented by nodes independently of the number of objects or complexity of the environment. We extensively analyze their contribution for human motion prediction. The results observed in different actions suggest that the models proposed are able to understand human activities significantly better than state-of-art models which do not use context, improving both human and object motion prediction.

### 6.4. Robustness to noise

All previous works on human motion prediction use ground truth MoCap data as past observations. Nevertheless, real applications will receive joint observations from e.g. human pose estimation models, such as OpenPose [7] or AlphaPose [13, 64], which are prone to suffer from noise and mis-detections, specially under strong occlusions. In this subsection, we therefore evaluate the resilience of our proposed models and previous baselines to noise in the input observations. Predictions are evaluated on the original ground truth data. The 3D coordinates of past observations (both in human and objects positions) are corrupted by additive Gaussian noise $\mathcal{N}(0, \sigma^2)$. In Table 3 we show the results of this experiment, with different values of $\sigma$. Interestingly, the error in the predictions gracefully increases with the noise, but still, our approach performs consistently better than those approaches that do not consider the context information. Indeed, the best context-aware models ($C\text{-}RNN+LI$ and $C\text{-}RNN+OMP+LI$) with noise up to $\sigma = 50\text{mm}$, perform better than context-less baselines with no noise in the input.

### 7. Conclusion

In this work, we explore a context-aware motion prediction architecture, using a semantic-graph representation where objects and humans are represented by nodes independently of the number of objects or complexity of the environment. We extensively analyze their contribution for human motion prediction. The results observed in different actions suggest that the models proposed are able to understand human activities significantly better than state-of-art models which do not use context, improving both human and object motion prediction.

### Acknowledgements

This work has been partially funded by the Spanish government under projects HuMoUR TIN2017-90086-R, ERA-Net Chistera project IPALM PCI2019-103386 and María de Maeztu Seal of Excellence MDM-2016-0656. We also thank Nvidia for hardware donation.

---

Table 3: Robustness to noise in Human and Object Motion Prediction. Average performance of the principal models when using the original test set (Noise-free input), compared to their performance when seeing noisy observations.

<table>
<thead>
<tr>
<th>Model</th>
<th>Noise-free input (HMP)</th>
<th>Noisy input (HMP)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>HMP Models 25 mm</td>
<td>50 mm</td>
</tr>
<tr>
<td></td>
<td>100 mm</td>
<td></td>
</tr>
<tr>
<td>Time (s)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.5</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>0.5</td>
</tr>
<tr>
<td>ZV [45]</td>
<td>61</td>
<td>150</td>
</tr>
<tr>
<td>RNN [43]</td>
<td>55</td>
<td>110</td>
</tr>
<tr>
<td>C-RNN+LI</td>
<td>52</td>
<td>104</td>
</tr>
<tr>
<td>C-RNN+OMP+LI</td>
<td>56</td>
<td>109</td>
</tr>
</tbody>
</table>

Table 4: Mean and Std of prediction errors (mm) on the CMU dataset. Our Context-aware model C-RNN+LI outperforms baselines even though context only consists of two people.

<table>
<thead>
<tr>
<th>Model</th>
<th>Time (s)</th>
<th>0.5</th>
<th>1.0</th>
<th>1.5</th>
<th>2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>µ</td>
<td>σ</td>
<td>µ</td>
<td>σ</td>
<td>µ</td>
</tr>
<tr>
<td>ZV</td>
<td>127</td>
<td>32</td>
<td>271</td>
<td>66</td>
<td>374</td>
</tr>
<tr>
<td>RNN</td>
<td>125</td>
<td>28</td>
<td>262</td>
<td>58</td>
<td>378</td>
</tr>
<tr>
<td>QuaterNet</td>
<td>138</td>
<td>26</td>
<td>279</td>
<td>58</td>
<td>378</td>
</tr>
<tr>
<td>C-RNN+LI</td>
<td>124</td>
<td>27</td>
<td>257</td>
<td>53</td>
<td>352</td>
</tr>
</tbody>
</table>
References

[31] Lucas Kovar, Michael Gleicher, and Frédéric Pighin. Motion graphs. In SIGGRAPH, 2008. 1

[34] CMU Graphics Lab. Cmu motion capture database. http://mocap.cs.cmu.edu/, 1, 2, 5

[35] Colin Lea, Austin Reiter, René Vidal, and Gregory D Hager. Segmental spatiotemporal cnns for fine-grained action segmentation. In ECCV, 2016. 2


[41] Priyanka Mandikal and Venkatesh Babu Radhakrishnan. Dense 3d point cloud reconstruction using a deep pyramid network. In WACV, 2019. 2


[46] Bingbing Ni, Xiaokang Yang, and Shenghua Gao. Progressively parsing interactional objects for fine grained action detection. In CVPR, 2016. 2

[47] Brian Paden, Michal Čáp, Sze Zheng Yong, Dmitry Yershov, and Emilio Frazzoli. A survey of motion planning and control techniques for self-driving urban vehicles. IV, 2016. 1


[52] Albert Pumarola, Jordi Sanchez-Riera, Gary Choi, Alberto Sanfelix, and Francisc Moreno-Noguer. 3dpeople: Modeling the geometry of dressed humans. In ICCV, 2019. 2


[57] Edgar Simo-Serra, Carme Torras, and Francisc Moreno-Noguer. 3d human pose tracking priors using geodesic mixture models. 2017. 2


[63] Jiayin Xie and Nilanjan Chakraborty. Rigid body motion prediction with planar non-convex contact patch. In ICRA, 2019. 2

