A New Spatio-Temporal Loss Function for 3D Motion Reconstruction and Extended Temporal Metrics for Motion Evaluation

Mansour Tchenegnon, Sylvie Gibet, Thibaut Le Naour

▶ **To cite this version:**

Mansour Tchenegnon, Sylvie Gibet, Thibaut Le Naour. A New Spatio-Temporal Loss Function for 3D Motion Reconstruction and Extended Temporal Metrics for Motion Evaluation. European Conference on Computer Vision (ECCV 2022), Workshop on What is Motion for?, Oct 2022, Tel Aviv, Israel. hal-03966941

HAL Id: hal-03966941
https://hal.science/hal-03966941v1

Submitted on 1 Feb 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A New Spatio-Temporal Loss Function for 3D Motion Reconstruction and Extended Temporal Metrics for Motion Evaluation

Mansour Tchenegnon[1], Sylvie Gibet[1], and Thibaut Le Naour[2]

[1] IRISA, Univ Bretagne Sud, Vannes, France
[2] Motion-Up Company, Vannes, France
{mansour.tchenegnon,sylvie.gibet}@univ-ubs.fr, tlenaour@motion-up.com

Abstract. We propose a new loss function that we call Laplacian loss, based on spatio-temporal Laplacian representation of the motion as a graph. This loss function is intended to be used in training models for motion reconstruction through 3D human pose estimation from videos. It compares the differential coordinates of the joints obtained from the graph representation of the ground truth against the one of the estimation. We design a fully convolutional temporal network for motion reconstruction to achieve better temporal consistency of estimation. We use this generic model to study the impact of our proposed loss function on the benchmarks provided by Human3.6M. We also make use of various motion descriptors such as velocity, acceleration to make a thorough evaluation of the temporal consistency while comparing the results to some of the state-of-the-art solutions.

Keywords: 3D human pose estimation, motion reconstruction, neural network, loss functions, Laplacian representation, evaluation metrics

1 Introduction

The 3D human pose estimation is a topic that has been extensively studied in recent years. The goal is to reconstruct the 3D skeletal pose from the image. One way to achieve this goal involves first to estimate the 2D joint locations from the image [5, 18]. Then, from these 2D joint locations, the 3D skeletal pose can be estimated. This process can be extended to video data. Given an input video or a sequence of 2D joint locations for each frame of the video, the objective becomes to compute the 3D joint positions for each frame. Working with video brings some advantages. In particular, adding temporal information can improve the learning of depth information [15]. Therefore, new solutions using temporal features from the input data are the subject of recent research.

Many deep learning methods learn from temporal information. Among these, previous methods based on Recurrent Neural Networks (RNN) have shown their efficiency but in return are very time consuming. Other more efficient solutions have then been proposed, based on convolutional neural networks [7, 19]. In this

case, the motion is considered as a temporal signal of joint positions on which convolution filters are applied. Some researchers propose to learn temporal features from adjacent frames through a convolution network, and then regress this information to estimate the central frame [17]. This solution improves the pose estimation but the temporal consistency of the motion is not necessarily better since the poses are still estimated one by one. In theory, the more accurate the pose estimation per frame, the more continuous the reconstructed motion. However, sufficient accuracy to obtain a temporally plausible reconstructed motion has not yet been achieved. Given this fact, we propose to use a convolutional neural network, called CVM-Net, to reconstruct motion through sequence-to-sequence pose estimation. The goal then becomes a motion reconstruction task. In this paper, the idea is not to outperform the spatial accuracy of recent neural network approaches, but to propose an approach that better preserves the temporal consistency.

Besides the design of this fully convolutional network, our main contribution is the definition of a new loss function for sequence-to-sequence pose estimation, called **the Laplacian Loss Function**. This function exploits a spatio-temporal representation of the poses sequence, inspired by the Laplacian graph $3D + t$ model [16]. Indeed, for this kind of sequence-to-sequence neural network, using only the default *Joint Position Loss* function (average Euclidean loss for each posture of the sequence), leads to the averaging of the posture errors over the whole motion sequence. This is achieved by filtering out extreme postures and by favouring those postures that are most representative of the training data. By focusing on the Laplacian representation of motion, we take into account both the spatial structure of each pose, constrained by the skeleton, and the temporal trajectories of the skeletal joints.

Our goal is to preserve the temporal consistency of the reconstructed motion first and then achieve an acceptable spatial accuracy. Therefore, we need adequate metrics for the evaluation of the neural network that consider the temporal characteristics of movements. On the spatial aspect, the Mean Per-Joint Positions Error (MPJPE) used in all state-of-the-art solutions is the best choice. Most approaches limit their evaluation to this metric since most of them focus only on achieving the best accuracy in reconstructing the joint positions. Very few approaches make a temporal consistency evaluation. For that, they use the Mean Per-Joint Velocity Error (MPJVE), that evaluates the estimated velocity error of the joints, computed on adjacent poses. This metric is a good start to evaluate the temporal consistency of a motion. However, motion characteristics are not limited to the velocity. We propose to extend the evaluation to the acceleration, so we propose a metric to evaluate the acceleration of the reconstructed motion, that is the **Mean Per-Joint Acceleration Error, MPJAccE**.

In this paper, we present in Section 3 the generic network for sequence-to-sequence estimation that we propose and the reasons for our choice. In Section 4, after presenting some traditional loss functions, we propose a new spatio-temporal loss function based on the Laplacian representation of the motion. Section 5 presents the extended metrics that we propose to evaluate the tem-

poral consistency of reconstructed motion. Finally in Section 6 we present and discuss the results of our experiments, including an ablation study for the **Laplacian loss** and the evaluation results based on metrics computed from motion descriptors.

2 Related Work

In this paper, we address the issue of reconstructing motion from video through 3D human pose estimation.

2.1 3D Human Pose Estimation

3D human pose estimation consists in estimating 3D skeletal poses given an image or 2D joint locations. According to the type of data used as input, we have two main categories. The first category uses the image as input and directly estimates 3D poses. In this case, the methods compute 2D and 3D features, such as heat maps and other features (camera focal length, depth information) to estimate the final 3D poses. These methods generally involve two stages: a features detection stage followed by a 3D pose estimation [24, 9, 12, 22]. Yang et al. [24] use a 3D estimation pose network and a pose discriminator to ensure that the estimated poses are plausible. Wei et al. [22] use a framework to generate heat maps and bone maps in order to extract 2D pose hypotheses. They then use a pose regressor or a selection-based algorithm on these hypotheses to compute the final 3D pose.

In the second category, one starts by estimating 2D joint locations in the image using a 2D pose estimator. From the estimated 2D joint locations, through various methods, it is possible to estimate the corresponding 3D poses. The main advantage of this approach is that it is more efficient on videos in the wild, due to the use of state-of-the-art 2D estimators. Some researchers propose lifting models [13, 3, 6, 26, 20, 27]. Martinez et al. [13] propose an approach using consecutive linear layers to perform a 2D-to-3D joint positions regression. Combining a 2D-to-3D pose regression and a 3D-to-2D pose re-projection modules, Biswas et al. [3] use information in . Chen et al. [6] present an unsupervised algorithm that lifts 2D joints to 3D skeletons. They show that adding random 2D projections and an adversarial network allows the training process to be self supervised using geometric consistency. Shimada et al. [20] decide to first estimate 3D pose from 2D joints locations, and then make the estimated pose more realistic, through foot contact prediction and physics-based pose optimization. Zou et al. [27] and Zhao et al. [26] represent the 2D joint locations as a graph structure and use a Graph Convolutional Networks to estimate the 3D pose from it. Azizi et al. [2] encode transformations between joints using the Möbius Transformation and propose a new light Spectral GCN to achieve state-of-the-art results. All these approaches focus on 3D pose regression and achieves great results in 3D pose estimation.

2.2 Motion Reconstruction

Network architecture Many 3D pose estimators are currently proposed in the literature. Most of them only work on one image at a time. When receiving a video as input, they estimate the pose at each frame, and then directly concatenate the outputs. This way of reconstructing a motion does not take into account the temporal characteristics of motion. This leads to some unsteady movements in the results, and very few approaches have considered these effects [14, 17, 4, 19, 21, 23, 7, 8]. Among them, Metha et al. [14] choose to infer the pose at time $t-1$ to estimate the pose at time t. Wang et al. [21] represent the 2D skeleton input as a spatio-temporal graph and propose a Graph Convolution Network to predict 3D poses. Xu et al. [23] choose to first estimate 3D poses and then use a trajectory completion framework to correct the sequence. More recently, Shi et al. [19] propose a CNN approach coupled with a skeleton model to correct the spatial joint positions. In their solution, two independent CNN models are first in charge of estimating the sequence of rotations and the bone lengths to preserve some of the skeleton constraints. From these features, they apply the forward kinematics model to obtain the sequence of 3D poses. Another solution proposed by Chen et al. [7] is to predict the length and direction of the bones throughout the sequence and compute the 3D poses from these. Our approach is based on temporal convolution to estimate 3D pose sequence from 2D joint locations sequence (obtained from video using a state-of-the-art 2D pose estimator).

Loss function Since motion reconstruction refers to spatio-temporal data, it is necessary to have an appropriate loss function to ensure a better learning process. The loss functions used in 3D human pose estimation are insufficient for this task because they focus on spatial accuracy alone. To solve this situation, some researchers propose new loss functions based on temporal characteristics of the motion. Among them, some choose to calculate the loss function by using the first derivative, that is the velocity [25, 4]. Wang et al. [21] propose a loss function, called *Motion Loss*, computed from a motion pose encoding space. They project the predicted and ground truth joint positions into this space and compute the difference between the two encoded information. This difference evaluates the quality of the reconstructed motion. Unlike them, we propose a spatio-temporal loss function as a solution to the learning process.

3 Deep Learning and motion temporal features

The human pose estimation task consists in estimating with accuracy the joints locations of the skeleton. Therefore, it mainly focuses on the spatial aspect of the motion, whether it uses a single frame or multiple frames. Our aim is to preserve the temporal aspect of the reconstructed motion, while achieving acceptable joint position errors.

Recent approaches prove that including temporal features of the motion improves the results in human pose estimation. A good technique to process temporal data is to use a RNN (Recurrent Neural Network). It allows to use information

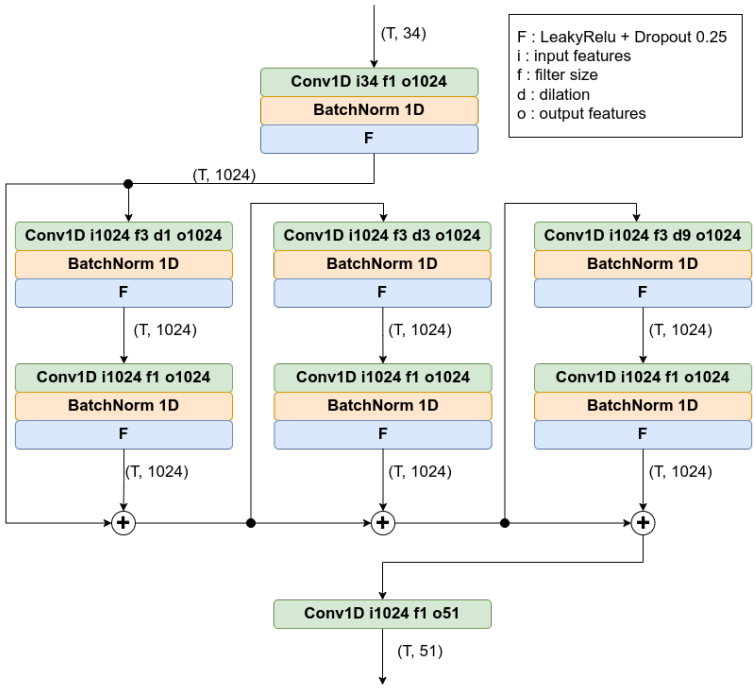

Fig. 1: CVM-Net architecture. Temporal Convolution Neural Networks for motion reconstruction. This is a generic approach that combines 1D convolution to transform the 2D poses sequence into a 3D poses sequence. It also learn long duration relation between joints via dilation parameters

from previous frames to estimate the current one. However the computation is time and resources consuming. To overcome this drawback, an efficient and less consuming solution is to use Convolutional Neural Networks (CNN). In a survey conducted by Kiranyaz et al. [11], some studies on 1D convolution networks prove that they have a low computational complexity and are well-suited for low-cost and real-time applications. Besides, Arsalan et al. [1] have also proven 1D convolution to be effective for trajectory-based air writing recognition. By considering a motion as a set of skeletal joint trajectories, 1D convolution is therefore a suitable choice to build our model.

1D convolution applies filters using a time window. It takes as input $b \times T_{in} \times C_{in}$ and output $b \times T_{out} \times C_{out}$, where b is the batch size, T_{in} and T_{out} represent respectively the sequence lengths of the input and output, and C_{in} and C_{out} the channels or features for input and output. Moreover, we can use a dilated convolution to apply filters on non-consecutive frames to learn features at different time scales. Another advantage of the fully convolutional neural network is that it does not require a fixed sequence length and as a result can be easily generalized.

Like many recent solutions, we have built our CVM-Net neural network to estimate poses through temporal convolution. By using multiple temporal convolution layers on frames in a time interval $[t - \frac{\tau}{2}, t + \frac{\tau}{2}]$, of length $\tau + 1$, the accuracy of the the central frame estimation t is improved. But unlike most of these approaches, our solution estimates multiple output frames (T_{out}) from multiple input frames (T_{in}), with $T_{in} = T_{out}$.

We used this neural network to study of the different functions we propose in this paper, namely the loss function and the motion evaluation metrics.

4 Loss Functions

The loss function is one of the main part of a neural network. This is a key element in the training of neural networks, which indicates to the model its erroneous behaviour and brings possible corrections. The loss function is chosen according to the task at hand. Existing loss functions for human pose estimation and motion reconstruction focus on either the spatial aspect or the temporal aspect of the movement. Both are computed separately and them combined. This can be a limitation since each aspect, spatial or temporal, is considered independently. It is therefore necessary to find the appropriate coefficients of the linear combination while computing the global loss. In this section, we propose a loss function based on both the temporal and spatial characteristics of the motion.

4.1 Existing Loss Functions

Joint Position Loss This is the most commonly used loss function in a single frame pose estimation context. It computes an average distance between the

joint positions of the ground truth poses and the estimated poses. Applied to multiple frames poses estimations it is defined as:

$$\mathcal{L}_P = \frac{1}{T} * \frac{1}{J} \sum_{t=1}^{T} \sum_{j=1}^{J} \|P_{t,j} - \overline{P}_{t,j}\|_2 \tag{1}$$

where $P_{t,j}$ and $\overline{P}_{t,j}$ are respectively the 3D estimated position and the 3D ground truth position of joint j at time t. This function, if used solely as loss function, works well for single frame pose estimation. But, when working on motion reconstruction, it is limited because it tends to average the joint positions loss over the whole sequence. The less represented poses in the motion can be biased by the more represented ones, affecting the overall motion reconstruction.

Motion Loss Wang et al. [21] propose a loss function as a distance in motion space. It is based on the encoding motion from a sequence of poses, by computing differential values between same joints at different time scales. It can be a subtraction, an inner-product or a cross-product. They encode both the estimated and the ground truth poses sequences. The loss is then computed between the encoded ground truth and reconstructed poses. They then combine this motion loss with the joint positions loss to compute an overall cost function defined by:

$$\mathcal{L} = \mathcal{L}_P + \lambda * \mathcal{L}_M \tag{2}$$

where $\mathcal{L}_M$ represents the motion loss, $\mathcal{L}_P$ the joint position loss and λ a coefficient to apply on the motion loss.

4.2 Laplacian Loss

We propose a loss function for spatio-temporal features learning, which is based on the Laplacian representation of motion as a $3D + t$ graph, as defined by Le Naour et al. [16]. The skeleton joints of the motion are considered as the nodes of the graph. The $3D + t$ graph is then obtained by i) first, connecting the joints to form the skeleton at each frame; these are the spatial edges. ii) We then create temporal edges by connecting joints between consecutive frames. Let's consider a motion of length T by a skeleton of J joints, and let $v_{j,t}$ be a node of the graph, representing the joint j of the skeleton at time t. We create the temporal edges by connecting $v_{j,t}$ to $v_{j,t-1}$ and $v_{j,t+1}$ which are the joints j of skeletons at time $t-1$ and $t+1$ respectively. The graph $3D+t$ is then defined by $G = (V, E_S \cup E_T)$, with $V = \{v_{j,t}\}$ the set of all the joints, E_S the set of spatial edges and E_T the set of temporal edges.

Using this spatio-temporal graph allows to define in an unified structure, the spatial and temporal relations between each joints of the sequence. From the graph 3D+t, we extract the matrix L. It is a square matrix of dimension $(N \times N)$ where $N = T * J$ represents the total number of joints (the number of frames T in the sequence is multiplied by the number of joints J per skeleton).

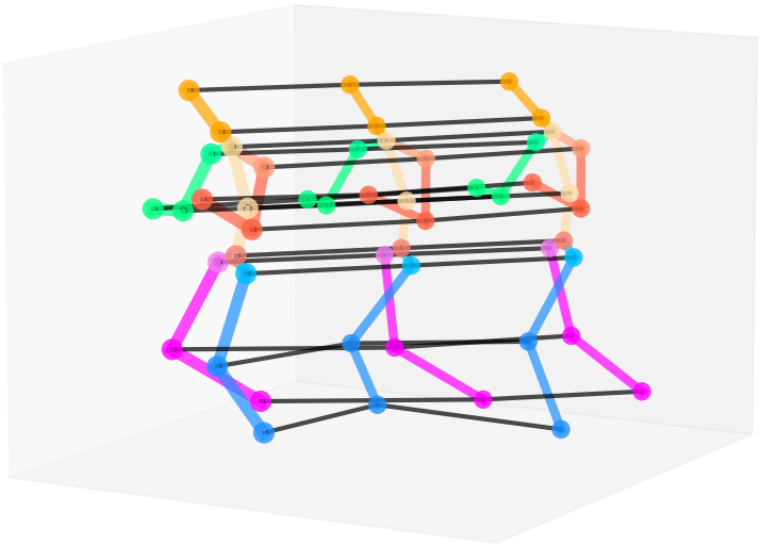

Fig. 2: Graph $3D+t$ of a motion of length 3 for a skeleton of 17 joints. The black edges represent the temporal edges of E_T while the coloured edges represent the spatial edges of E_S.

$w_{i,j}$ represents the weight attributed to each edge of the graph. In our case we apply uniform weights for both spatial and temporal edges.

$$w_{i,j} = \begin{cases} 1 \text{ if } (i,j) \in E_S \text{ or } E_T \\ 0 \text{ otherwise} \end{cases} \tag{3}$$

Let's define the matrix Δ, so called *differential coordinates matrix*, computed from the Laplacian matrix L and the joint positions matrix P is of dimension $(N \times 3)$: $\Delta = L * P$.
Δ is then of dimension $(N \times 3)$. The matrix Δ represents the differential coordinates of each joint relatively to its neighbours. The closer the differential coordinates of the estimated poses to those of the ground truth, the better the reconstructed motion. We can therefore compute an average distance error between the differential coordinates of the ground truth and those of the estimation. This distance represents our loss value computed as follows:

$$\mathcal{L}_\Delta = \frac{1}{N} \sum_{1}^{N} \|\Delta^{gt} - \bar{\Delta}\| \tag{4}$$

where $N = T.J$ is the total number of joints with T the sequence length and J the number of joints for a skeleton, Δ^{gt} is the matrix of differential coordinates computed from the ground truth and $\bar{\Delta^{gt}}$ the matrix of differential coordinates computed from the estimation. By training the neural network with such a loss

function will allow to learn the connections between the joints locations it predicts, thus taking into account implicitly the skeletal structure and the temporal evolution of the joints.

$\mathcal{L}_\Delta$ computes the mean absolute error between the differential coordinates extracted from the ground truth joint positions and the estimated joint positions. This loss is based on a differential representation of the original output and does not consider the absolute joint positions. Therefore we combine it with the joint positions loss to compute an overall loss:

$$\mathcal{L} = \mathcal{L}_P + \alpha * \mathcal{L}_\Delta \tag{5}$$

where $\mathcal{L}_P$ is the joint positions loss, $\mathcal{L}_\Delta$ is the Laplacian loss and α is the coefficient applied on the Laplacian loss.

5 Evaluation Metrics for Temporal Consistency Performance

There are two existing metrics for evaluating human pose estimation. The main metric used in the state of the art is the Mean Per Joint Position Error (MPJPE, same formula as the Joint Position Loss 1). It evaluates the spatial accuracy of the models, using the average errors in estimating the joint positions. The lower the error, the better the accuracy. The Mean Per Joint Velocity Error, on the other hand, evaluates the temporal consistency by computing the average velocity error between the estimation and the ground truth. In the state-of-the-art, it is computed as defined in equation 7. Temporal consistency refers to how close the reconstructed motion is to the ground truth in terms of smoothness, velocity, or acceleration.

$$v_{j,t} = P_{j,t+1} - P_{j,t} \tag{6}$$

$$MPJVE = \frac{1}{T-1} * \frac{1}{J} \sum_{t=1}^{T-1} \sum_{j=1}^{J} \|v_{t,j} - \overline{v}_{t,j}\|_2 \tag{7}$$

where $v_{t,j}$ and $\overline{v}_{t,j}$ represent the velocity vectors of joint j at time t, respectively from the ground truth and the estimation.

In order to extend the temporal consistency evaluation to other motion descriptors. Therefore we define $MPJAccE$ metric (Mean Per Joint Acceleration Error in equation 9), which is based on acceleration.

$$a_{j,t+1} = P_{j,t+2} - 2 * P_{j,t+1} + P_{j,t} \tag{8}$$

$$MPJAccE = \frac{1}{T-2} * \frac{1}{J} \sum_{t=1}^{T-2} \sum_{j=1}^{J} \|a_{t,j} - \overline{a}_{t,j}\|_2 \tag{9}$$

where $a_{t,j}$ and $\overline{a}_{t,j}$ represent the acceleration vectors of joint j at time t, respectively from the ground truth and the estimation.

6 Experiments

6.1 Ablation Study

To evaluate the performance of our new loss function, we first set up training-test experiments using the neural network architecture proposed in section 3. We use the same training environment for each session. In these experiments, we compare three configurations of loss functions.

- **CVM-Net** uses only the joint position loss $\mathcal{L}_P$ as cost function (baseline).
- **CVM-Net + $\mathcal{L}_M$** uses a combination of the joint positions loss $\mathcal{L}_P$ and the *Motion Loss* $\mathcal{L}_M$ as proposed by Wang et al. [21] in an overall function.
- **CVM-Net + $\mathcal{L}_\Delta$** finally uses a combination of the joint positions loss and our *Laplacian Loss* $\mathcal{L}_\Delta$ in an overall function.

Table 1: Comparison of the three loss functions: CVM-Net (baseline), CVM-Net + $\mathcal{L}_M$, CVM-Net + $\mathcal{L}_\Delta$, with the reconstructed errors MPJPE (under Protocol-1) and MPJVE. Protocol-1 computes the MPJPE from joint positions relative to the root joint (central hip) by aligning the root joints of both the estimation and the ground truth. a) Comparison of loss functions with MPJPE; b) with MPJVE. For both metrics, the lower the better.

(a) MPJPE comparison results in mm.

MPJPE	Dir.	Dis.	Eat.	Greet.	Phon.	Phot.	Pos.	Purch.
CVM-Net	85.25	129.41	109.17	101.22	117.17	137.31	86.12	293.37
CVM-Net w/ $\mathcal{L}_M$	83.67	107.73	118.72	95.51	113.39	131.98	82.64	221.00
CVM-Net w/ $\mathcal{L}_\Delta$	**80.77**	**82.53**	**104.96**	**87.07**	**101.80**	**107.00**	**77.41**	**98.85**

MPJPE	Sit.	SitD.	Smok.	Wait.	WalkD.	Walk.	WalkT.	Avg
CVM-Net	152.75	248.98	119.87	105.45	261.62	87.20	87.81	142.47
CVM-Net w/ $\mathcal{L}_M$	148.74	232.07	113.56	95.76	195.00	85.04	83.95	127.99
CVM-Net w/ $\mathcal{L}_\Delta$	**137.33**	**178.99**	**103.43**	**84.37**	**104.32**	**79.16**	**76.30**	**100.62**

(b) MPJVE comparison results in mm/frame

MPJVE	Dir.	Dis.	Eat.	Greet.	Phon.	Phot.	Pos.	Purch.	Sit.
CVM-Net	3.34	4.63	3.39	4.73	3.11	3.87	3.17	9.21	2.78
CVM-Net w/ $\mathcal{L}_M$	3.25	4.73	3.15	4.44	2.91	3.84	2.95	11.39	2.54
CVM-Net w/ $\mathcal{L}_\Delta$	**2.88**	**2.94**	**2.59**	**3.58**	**2.29**	**2.79**	**2.60**	**3.55**	**1.94**

MPJVE	SitD.	Smok.	Wait.	WalkD.	Walk.	WalkT.	Avg
CVM-Net	4.92	3.15	3.55	6.86	5.71	4.84	4.50
CVM-Net w/ $\mathcal{L}_M$	4.64	2.87	3.29	6.83	5.01	4.22	4.42
CVM-Net w/ $\mathcal{L}_\Delta$	**3.22**	**2.25**	**2.56**	**4.13**	**4.20**	**3.60**	**3.01**

The study is achieved with the benchmark Human3.6M [10] that contains millions of frames of captured data. For each configuration previously defined, we evaluate the MPJPE (through Protocol-1) and MPJVE using 2D joint locations ground truth as input (obtained from the benchmark). The evaluation consists of using the full dataset for the training-evaluation experiments. The dataset is split into a training set and a test set according to the subjects whose movements were captured (subjects 1, 5, 6 and 7 for training and subjects 9, 11 for evaluation).

Table 1 presents the comparative results obtained with the different loss functions within the framework of Protocol 1 of the Human3.6M benchmark. Compared to the baseline, we observe an average decrease of $14.48mm$ when we add the Motion Loss $\mathcal{L}_M$ in the training process. In addition, the Laplacian loss $\mathcal{L}_\Delta$ provides a significant improvement of $41.85mm$ (average decrease in MPJPE). This shows that combining the joint positions loss with our Laplacian loss significantly improves accuracy. It also minimizes the velocity error by an average of $1.49mm/frame$. This shows the efficiency of our spatio-temporal loss function that combines both the spatial relationships (between joints of the same skeleton) and the temporal relationships (between joints of consecutive frames).

Some visualisation results are displayed in Figure 3.

6.2 Evaluation Metrics Study

Usually, human pose estimators are evaluated with the MPJPE metric which calculates the average error of the joint positions for each pose. Very few are evaluated with the MPJVE metric, which calculates the error on the velocity of movement. In our case, since we focus on the temporal aspect of motion, we added to MPJPE and MPJVE, a new acceleration-based metric, MPJAccE (see section 5). We compare our approach with some state-of-the art solutions using these three metrics. We chose three different types of approaches based on their input and output settings (sequence-to-sequence, sequence-to-pose, pose-to-pose). First the approach of Pavllo et al. [17] is a sequence-to-pose approach from which we derived ours. The second solution from Shi et al. [19] is one of the best sequence-to-sequence approaches in motion reconstruction that makes use of forward kinematics. The knowledge of the skeleton constraints (angle limits, bones lengths) is embedded in this model. Finally, the solution from Zhao et al. [26] is a pose-to-pose approach that achieves good results on 3D human pose estimation. Table 2 shows the results obtained during our evaluation. Our approach using $\mathcal{L}_\Delta$, although far behind the state-of-the art results for MPJPE (2.5× less accurate), achieves good results for MPJVE and MPJAccE. The MPJVE results show that our proposal is comparable to the other solutions, although it is a generic approach. We achieve the best results for the MPJAccE metric. Since our approach does not take into consideration the preservation of skeletal bone lengths, unlike other methods, it is normal that the MPJPE measurement is unfavourable to us. As for the MPJVE metric, which calculates the vector of changes in position (both in velocity and direction), it filters in a certain way the joint positions. The acceleration (second order derivative), which characterizes the variations of the velocity, reflects another measure of motion

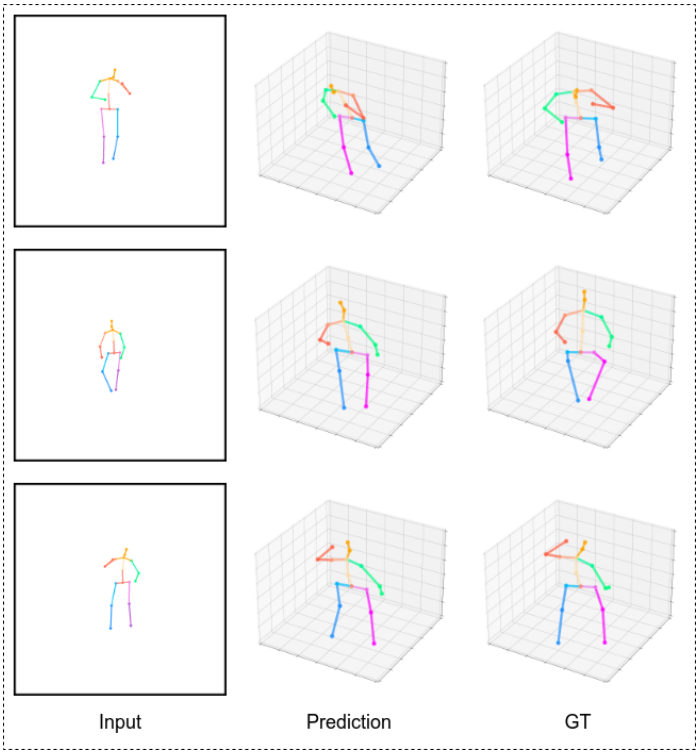

Fig. 3: Visualisation results of CVM-Net + $\mathcal{L}_\Delta$ on Human3.6m. On the left the input as 2D joint locations. On the middle our prediction and on the right the ground truth.

features that filters the velocity. The results on the two metrics MPJVE and MPJAccE show that our method better preserves the temporal characteristics of the original motion than other approaches.

7 Conclusion

In this paper we have presented a new spatio-temporal loss function based on the representation of the motion as a 3D+t graph. We have shown that this function improves both the spatial accuracy and the temporal consistency of 3D sequence-to-sequence pose estimation for motion reconstruction. We have used a temporal convolutional neural network for sequence-to-sequence pose estimation on a large scale dataset Human3.6M. Although this generic model does not challenge state-of-the art solutions on spatial accuracy with MPJPE evaluation, it has proven the efficiency of the Laplacian Loss in the spatio-temporal encoding of motion and the improvement of the temporal consistency. Moreover, such a solution is to be preferred in a situation where we are interested in the differential features

(speed, acceleration), for the in-depth analysis of the movement for example. Based on these preliminary results, our future work will integrate this Laplacian loss with other strategies – including bone length constraints – to improve the spatial accuracy while preserving the current temporal consistency. We will also integrate this loss function into the training of existing state-of-the-art methods in order to further validate its efficiency and improve the results, both on spatial and temporal aspects. Finally, we intend to propose a post-processing method based on the Laplacian representation to correct the results of the methods having obtained the best scores (spatial accuracy) and thus to obtain a better temporal consistency.

In this work we also have proposed a new protocol for evaluating temporal consistency in motion reconstruction through 3D sequence-to-sequence pose estimation. Velocity and acceleration measurements provide metrics that extend the classical position metric and allow to evaluate different solutions according to criteria related to the temporal quality of motion.

Table 2: Comparison with state-of-the-art solutions. a) Comparison with MPJPE under Protocol-1, b) MPJVE comparison, c) MPJAccE comparison. Best results are in bold. Legend : (‡) sequence-to-sequence approach, (†) sequence-to-pose approach with multi-frames as input and single frame as output, (∗) pose-to-pose approach, f=frame

(a) MPJPE comparison results in mm.

MPJPE	Dir.	Dis.	Eat.	Greet.	Phon.	Phot.	Pos.	Purch.
Shi et al. [19] (‡)	45.48	51.28	49.43	51.91	52.58	66.46	50.59	48.46
Pavllo et al. [17] (†)	**33.88**	43.99	44.28	48.96	44.62	65.80	**32.79**	55.12
Zhao et al. [26] (∗)	38.62	**43.08**	**35.89**	**40.15**	**40.85**	**50.14**	42.56	**40.40**
Ours(+$\mathcal{L}_\Delta$) (‡)	80.78	82.50	104.44	87.03	101.16	106.88	77.43	98.24

MPJPE	Sit.	SitD.	Smok.	Wait.	WalkD.	Walk.	WalkT.	Avg
Shi et al. [19] (‡)	55.90	64.25	53.79	52.84	58.85	49.99	48.25	53.47
Pavllo et al. [17] (†)	**45.61**	**48.09**	57.29	47.09	45.16	43.30	46.67	46.84
Zhao et al. [26] (∗)	47.81	56.47	**42.20**	**42.25**	**42.29**	**33.39**	**36.00**	**42.14**
Ours(+$\mathcal{L}_\Delta$) (‡)	136.73	178.48	102.98	84.33	103.65	79.24	76.39	100.34

(b) MPJVE comparison results in mm/f

MPJVE	Dir.	Dis.	Eat.	Greet.	Phon.	Phot.	Pos.	Purch.
Shi et al. [19] (‡)	3.08	3.38	2.41	3.64	2.39	3.41	2.71	2.87
Pavllo et al. [17] (†)	2.78	**2.42**	3.10	3.72	2.68	2.87	3.12	**2.71**
Zhao et al. [26] (∗)	**2.57**	2.84	**2.40**	**3.41**	**2.14**	**2.60**	**2.56**	2.92
Ours(+$\mathcal{L}_\Delta$) (‡)	2.89	2.93	2.58	3.58	2.29	2.79	2.60	3.54

MPJVE	Sit.	SitD.	Smok.	Wait.	WalkD.	Walk.	WalkT.	Avg
Shi et al. [19] (‡)	**1.53**	2.19	2.44	2.69	5.56	4.43	4.13	3.12
Pavllo et al. [17] (†)	3.43	2.27	2.07	**2.31**	**2.98**	**2.24**	**3.11**	2.79
Zhao et al. [26] (∗)	1.57	**2.18**	**2.02**	2.53	3.83	4.02	3.49	**2.74**
Ours(+$\mathcal{L}_\Delta$) (‡)	1.94	3.22	2.25	2.56	4.13	4.19	3.59	3.01

(c) MPJAccE comparison results in mm/f^2

MPJAccE	Dir.	Dis.	Eat.	Greet.	Phon.	Phot.	Pos.	Purch.
Shi et al. [19] (‡)	1.87	2.22	1.26	2.04	1.51	2.18	1.43	**1.52**
Pavllo et al. [17] (†)	2.33	2.05	2.47	2.76	2.13	2.88	2.56	2.26
Zhao et al. [26] (∗)	1.74	1.94	1.61	2.40	1.40	1.81	1.64	2.15
Ours(+$\mathcal{L}_\Delta$) (‡)	**1.12**	**1.21**	**1.03**	**1.42**	**0.98**	**1.23**	**1.06**	1.83

MPJAccE	Sit.	SitD.	Smok.	Wait.	WalkD.	Walk.	WalkT.	Avg
Shi et al. [19] (‡)	**0.66**	**1.04**	1.50	1.58	4.57	3.17	2.90	1.96
Pavllo et al. [17] (†)	2.72	2.05	2.09	2.07	2.34	**1.81**	2.64	2.34
Zhao et al. [26] (∗)	1.02	1.52	1.23	1.64	2.83	3.01	2.34	1.89
Ours(+$\mathcal{L}_\Delta$) (‡)	0.88	1.91	**0.88**	**1.03**	**1.91**	1.93	**1.50**	**1.33**

References

1. Arsalan, M., Santra, A., Issakov, V.: Radar trajectory-based air-writing recognition using temporal convolutional network. pp. 1454–1459. Institute of Electrical and Electronics Engineers Inc. (12 2020). https://doi.org/10.1109/ICMLA51294.2020.00225
2. Azizi, N., Possegger, H., Rodol, E., Bischof, H.: 3d human pose estimation using mbius graph convolutional networks. ArXiv (2022)
3. Biswas, S., Sinha, S., Gupta, K., Bhowmick, B.: Lifting 2d human pose to 3d : A weakly supervised approach. pp. 1–9 (5 2019)
4. Cai, Y., Ge, L., Liu, J., Cai, J., Cham, T.J., Yuan, J., Thalmann, N.M.: Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks. Proceedings of the IEEE International Conference on Computer Vision **2019-Octob**, 2272–2281 (2019). https://doi.org/10.1109/ICCV.2019.00236
5. Cao, Z., Hidalgo, G., Simon, T., Wei, S.E., Sheikh, Y.: Openpose: Realtime multi-person 2d pose estimation using part affinity fields. Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017 **2017-Janua**, 1302–1310 (12 2018), http://arxiv.org/abs/1812.08008
6. Chen, C.H., Tyagi, A., Agrawal, A., Drover, D., MV, R., Stojanov, S., Rehg, J.M.: Unsupervised 3d pose estimation with geometric self-supervision. pp. 5707–5717 (4 2019)
7. Chen, T., Fang, C., Shen, X., Zhu, Y., Chen, Z., Luo, J.: Anatomy-aware 3d human pose estimation with bone-based pose decomposition. IEEE Transactions on Circuits and Systems for Video Technology **32**, 198–209 (2022). https://doi.org/10.1109/TCSVT.2021.3057267
8. Dabral, R., Mundhada, A., Kusupati, U., Afaque, S., Sharma, A., Jain, A.: Learning 3d human pose from structure and motion. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) **11213 LNCS**, 679–696 (2018)
9. Habibie, I., Xu, W., Mehta, D., Pons-Moll, G., Theobalt, C.: In the wild human pose estimation using explicit 2d features and intermediate 3d representations. pp. 10897–10906 (4 2019)
10. Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. IEEE Transactions on Pattern Analysis and Machine Intelligence **36**, 1325–1339 (2014). https://doi.org/10.1109/TPAMI.2013.248
11. Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M., Inman, D.J.: 1d convolutional neural networks and applications: A survey. Mechanical Systems and Signal Processing **151**, 107398 (4 2021). https://doi.org/10.1016/j.ymssp.2020.107398
12. Luvizon, D.C., Picard, D., Tabia, H.: 2d/3d pose estimation and action recognition using multitask deep learning. pp. 5137–5146. IEEE (2018). https://doi.org/10.1109/CVPR.2018.00539
13. Martinez, J., Hossain, R., Romero, J., Little, J.J.: A simple yet effective baseline for 3d human pose estimation. Proceedings of the IEEE International Conference on Computer Vision **2017-Octob**, 2659–2668 (2017). https://doi.org/10.1109/ICCV.2017.288
14. Mehta, D., Rhodin, H., Casas, D., Fua, P., Sotnychenko, O., Xu, W., Theobalt, C.: Monocular 3d human pose estimation in the wild using improved cnn supervision (2017)

15. Mehta, D., Sridhar, S., Sotnychenko, O., Rhodin, H., Shafiei, M., peter Seidel, H., Casas, D., Theobalt, C., peter Seidel, H., Xu, W.: Vnect real-time 3d human pose estimation with a single rgb camera. ACM Transactions on Graphics (SIGGRAPH 2017) **36**, 1–13 (2017)
16. Naour, T.L., Courty, N., Gibet, S.: Spatiotemporal coupling with the 3d+t motion laplacian. Computer Animation and Virtual Worlds **24**, 419–428 (2013). https://doi.org/https://doi.org/10.1002/cav.1518, https://onlinelibrary.wiley.com/doi/abs/10.1002/cav.1518
17. Pavllo, D., Feichtenhofer, C., Grangier, D., Auli, M.: 3d human pose estimation in video with temporal convolutions and semi-supervised training. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition **2019-June**, 7745–7754 (2019). https://doi.org/10.1109/CVPR.2019.00794
18. Rogez, G., Weinzaepfel, P., Schmid, C.: Lcr-net++: Multi-person 2d and 3d pose detection in natural images. IEEE Transactions on Pattern Analysis and Machine Intelligence **42**, 1146–1161 (2020). https://doi.org/10.1109/TPAMI.2019.2892985
19. Shi, M., University, S., ABERMAN, B.F.A.K., Aberman, K., Aristidou, A., Komura, T., Lischinski, D.: Motionet: 3d human motion reconstruction from monocular video with skeleton consistency. ACM Transactions on Graphics **40** (2020). https://doi.org/10.1145/3407659, https://doi.org/10.1145/3407659
20. Shimada, S., Golyanik, V., Xu, W., Theobalt, C.: Physcap: Physically plausible monocular 3d motion capture in real time. Real ACM Trans. Graph **39**, 16 (2020). https://doi.org/10.1145/3414685.3417877, https://doi.org/10.1145/3414685.3417877
21. Wang, J., Yan, S., Xiong, Y., Lin, D.: Motion guided 3d pose estimation from videos. European Conference on Computer Vision (ECCV) pp. 764–780 (2020)
22. Wei, G., Wu, S., Tang, K., Li, G.: Bonenet: Real-time 3d human pose estimation by generating multiple hypotheses with bonemap representation. Computer-Aided Design and Applications **18**, 1448–1465 (2021). https://doi.org/10.14733/cadaps.2021.1448-1465, https://doi.org/10.14733/cadaps.2021.1448-1465
23. Xu, J., Yu, Z., Ni, B., Yang, J., Yang, X., Zhang, W.: Deep kinematics analysis for monocular 3d human pose estimation. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition pp. 896–905 (2020). https://doi.org/10.1109/CVPR42600.2020.00098
24. Yang, W., Ouyang, W., Wang, X., Ren, J., Li, H., Wang, X.: 3d human pose estimation in the wild by adversarial learning. pp. 5255–5264 (3 2018)
25. Zhang, J., Tu, Z., Yang, J., Chen, Y., Yuan, J.: Mixste: Seq2seq mixed spatio-temporal encoder for 3d human pose estimation in video. ArXiv **abs/2203.00859** (2022), http://arxiv.org/abs/2203.00859
26. Zhao, L., Peng, X., Tian, Y., Kapadia, M., Metaxas, D.N.: Semantic graph convolutional networks for 3d human pose regression. vol. 2019-June, pp. 3420–3430. IEEE Computer Society (6 2019). https://doi.org/10.1109/CVPR.2019.00354
27. Zou, Z., Tang, W.: Modulated graph convolutional network for 3d human pose estimation. IEEE/CVF International Conference on Computer Vision (ICCV) pp. 11477–11487 (2021), https://github.com/