Temporal Shape Transfer Network for 3D Human Motion

João Regateiro, Edmond Boyer

Temporal Shape Transfer Network for 3D Human Motion

João Regateiro Edmond Boyer
Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP*, LJK, 38000 Grenoble, France
name.surname@inria.fr

Abstract

This paper presents a learning-based approach to perform human shape transfer between an arbitrary 3D identity mesh and a temporal motion sequence of 3D meshes. Recent approaches tackle the human shape and pose transfer on a per-frame basis and do not yet consider the valuable information about the motion dynamics, e.g.*, body or clothing dynamics, inherently present in motion sequences. Recent datasets provide such sequences of 3D meshes, and this work investigates how to leverage the associated intrinsic temporal features in order to improve learning-based approaches on human shape transfer. These features are expected to help preserve temporal motion and identity consistency over motion sequences. To this aim, we introduce a new network architecture that takes as input successive 3D mesh frames in a motion sequence and which decoder is conditioned on the target shape identity. Training losses are designed to enforce temporal consistency between poses as well as shape preservation over the input frames. Experiments demonstrate substantially qualitative and quantitative improvements in using temporal features compared to optimization-based and recent learning-based methods.*

1. Introduction

Motion retargeting is the process of transferring motion between digital characters. It is primarily used to create animations based on information captured on real characters and can therefore enrich creative applications as well as expand existing moving body datasets, *e.g.*, [6, 7, 24, 21].

Motion retargeting was originally performed using partial shape observations, as obtained with marker-based motion capture systems, and through skeletal parametrization of the body pose, *e.g.*, [19, 2, 15]. With the progress of computer vision algorithms and performance capture systems, it has since been extended to full shape observations, *e.g.*, [23, 3], with the objective to enable motion transfer with fully captured shape information. Such an extension gives access to more realistic body models, but faces an inherent difficulty with traditional skeletal parametrizations. In fact, while efficient, these parametrizations are prone to artifacts and unrealistic surface deformation as they introduce an intermediate pose representation that tends to be inaccurate with real characters. Surface-based approaches, such as [23, 8], were proposed as an alternative solution that directly deforms the surface of a character given another character surface in a different pose. Although they relax the need for skeletal parametrization, these approaches still build on the assumption that body poses can be modeled independently of body shapes. This seems difficult in practice as poses can hardly be formally identified in a consistent way among different real characters.

More recently, data-driven strategies have been proposed, *e.g.*, [4, 25, 28, 9], that demonstrate better pose and shape disentanglement abilities and hence better generalization. In particular [4, 9] propose to consider shape identities in the motion transfer, whereas most previous methods fo-

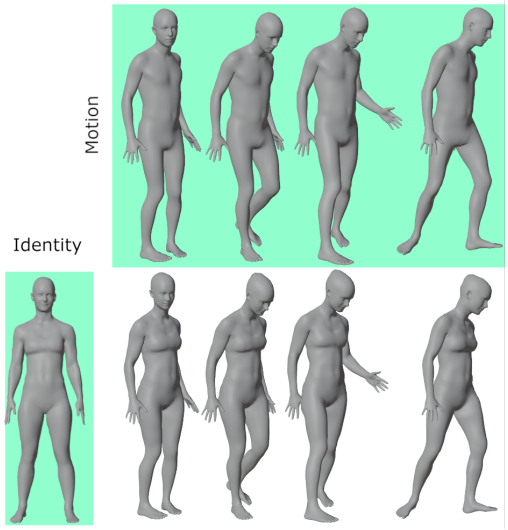

Figure 1. Given (in green) a shape identity and a shape motion sequence our spatio-temporal approach transfers the identity onto the motion sequence yielding a new sequence (bottom row).

*Institute of Engineering Univ. Grenoble Alpes

cus on pose transfer. They demonstrate thereby state-of-the-art results, and we build on this idea that we explore further with the temporal dimension. Leading motion retargeting methods transfer shape or pose features in a single frame, hence a static basis. However, moving shape observations are inherently temporal shape functions, a property that a static approach can not exploit. Temporal features that encode information on shape dynamics can provide additional and beneficial information for shape transfer, such as local shape features that are stable over time.

To this aim, we propose Temporal Shape Transfer Network, a data-driven approach to modify the identity of a moving character in a captured sequence of 3D meshes. The network is an autoencoder that considers consecutive frames in the input sequence in addition to an identity mesh. The encoder aggregates spatial features over time while the decoder combines identity shape features to the encoded latent motion representations. A discriminator is introduced to enforce the decoder to generate realistic shapes, which appears to be crucial, especially when dealing with unknown identities. To the best of our knowledge, this is the first learning-based approach that exploits temporal features on the problem of shape transfer. Experimental results with 4 consecutive frames already confirm their benefit with a significant improvement over the state-of-the-art in motion prediction and shape preservation for long sequences.

2. Related Work

Motion transfer methods can be divided into three main categories, as detailed below. Note that we focus on works most relevant to our concern with human shapes.

2.1. Skeletal Motion Transfer

As mentioned before, early strategies for motion retargeting are built on skeletal parametrizations, for instance, joint locations of the human pose. These parameterizations allow the transfer of motion between rigged meshes, the surfaces to which skeletal structures are attached through skinning weights [27, 19]. Surface deformations are then performed using blending techniques [17, 14], with the objective to match target poses. While popular for animating digital characters given synthesized or captured joint locations, this approach is prone to unrealistic surface deformations [2, 15], especially with real characters. In addition, it generally considers the transfer of new poses onto known identities and is less adapted to the transfer of new captured identities onto exiting motion sequences [26, 1], hence lacking generalization ability over human shapes.

2.2. Shape Motion Transfer

Omitting the intermediate skeletal parametrizations, surface-based approaches were later proposed to consider shape deformations directly, as with mesh deformations and through optimization-based methods [23, 3, 29, 22, 8, 10]. For instance, Sumner *et al*. and Baran *et al*. [23, 29] encode the pose of a source character as the deformation of the associated mesh and transfer it to a target character through per-triangle affine transformations, therein assuming complete mesh correspondences. This seminal work has inspired several approaches but still suffers from artifacts when transferring between significantly different shapes. Other works have explored semantic deformation transfer between characters allowing thus for very distinct shapes as in [3]. These methods usually exploit semantic pose correspondences that can be difficult to obtain in practice. Also, exploiting such given pose correspondences [8] investigates the animation of captured characters over time sequences instead of single frames. All these methods cast the problem of motion transfer as a pose transfer, assuming the implicit independence between pose and shapes. In contrast, Basset *et al*. [5] presents an optimization-based method that departs from the pose transfer paradigm and explores the transfer of shape identities instead. However, we follow a similar strategy with a learning-based approach that applies to sequences instead of single frames. The experiment section provides comparisons with [5].

2.3. Learning-based Motion Transfer

More recently, neural networks have proven efficient in learning motion transfer by training over examples of source and target poses [10, 28]. They have also demonstrated the ability to learn mappings between semantically different poses of humans and animals in order to interactively control animation generation [22]. Interestingly, these methods neither require skeletons nor point-to-point correspondences between source and target. However, heavy pre-processing needs to be performed for every pair of source and target characters. Moreover, learning-based methods usually fail to generalize to unseen examples and must be re-trained to handle unknown identities [28], whereas we propose to learn multiple shapes with a base template that allows generalization over new identities. Also, most learning-based approaches focus on transferring pose deformation between characters [25, 28], which can lead to unrealistic shape deformation with unseen poses and limits, therefore the range of admissible motions. Besides, Wang *et al*. [25] successfully proposed to reuse concepts from 2D image style transfer to learn a spatial adaptive network that is invariant to vertex order. The method can suffer from stretching artifacts when the identity shape has body contacts or limbs in proximity, often occurring with realistic captured data and over long sequences. More recently, Chen *et al*. [9] proposed to exploit the framework from [25] to disentangle pose and shape in an unsupervised manner. This approach contributes to the shape transfer problem with an unsupervised strategy, and

we consider another aspect with the temporal dimension and the ability to exploit existing datasets with dynamic information.

Our approach evolves from these previous works to tackle the inverse problem, where the posed shape deforms to adopt the style of the identity shape, avoiding, therefore, the issue of pose transfer between very distinct poses while maintaining generalization and vertex permutation invariant properties. The following section focuses solely on generalization and omits vertex permutation, which is inherent in the state-of-the-art methods [25]. In addition, our method can robustly transfer unseen shapes to long mesh sequences and hence suffers less from stretching or unrealistic artifacts. Comprehensive comparisons with the state-of-the-art are provided to validate these aspects.

3. Method

Our method considers temporal sequences of 3D meshes with the same connectivity, similarly to the state-of-the-art approaches [5, 25, 28]. However, in contrast to the frame-by-frame strategy usually followed, our learning-based method takes as input several consecutive frames that are fed into the network we propose, as illustrated in Figure 2. To take advantage of the temporal information, our network first extracts combined spatial and temporal features from the input mesh frames using the encoder presented in Section 3.1. Given the temporal features and a shape identity, the decoder transfers the shape identity onto the input frames while preserving its intrinsic and motion features over time as detailed in Section 3.2. The resulting shape deformations can be unrealistic with unseen identities (*i.e.*, not in the training set), and hence we introduce a discriminator to complete the architecture to enforce shape realism (Section 3.3). Our experiments (Section 4) validate this architecture and demonstrate the benefits of temporal features over traditional optimisation-based [5] and learning-based [4, 25, 28] methods for shape transfer.

3.1. Temporal Features

The objective is to transfer an identity onto a temporal shape sequence of a moving human body. In order to exploit the time dimension for that purpose, we expect our network to extract motion features from the input sequence that can, in turn, help the decoder generate time consistent deformations. To this aim, we use an LSTM recurrent neural network [13] as it has been successfully used for classification, processing and prediction based on temporal data. Of particular interest for this work is the LSTM's ability to handle either a single or an input sequence to learn dependencies within the sequence without significantly impacting computational resources.

The temporal encoder E_t is therefore a multi-layer LSTM network that takes as input 4 consecutive frames from a 3D mesh sequence, and outputs hidden state vectors c_t for each of the input frames (see Figure 2). These hidden state vectors are spatially correlated through time and, consequently, can compensate for the inability of point based architecture alone to exploit local surface information, as illustrated with body contact in Figure 3 with NPT (max-pool). Additionally, they help better preserving the motion dynamics present in the input sequence, as illustrated in Figures 5, 7 and 4.

Each hidden state c_t is concatenated with the feature vector s_t from the spatial encoder resulting in a temporal feature vector z_t. This feature is thus the latent representation of a sequence of posed meshes through time, which the decoder G receives as input.

3.2. Shape Preservation

When transferring an identity shape onto a moving shape sequence, it is desirable to preserve the identity shape features, or intrinsic features, throughout the sequence as well as to retain the original motion sequence, or extrinsic features, for each frame. The following sections describe the intrinsic and extrinsic metrics in more detail.

3.2.1 Temporal Intrinsic Shape Preservation

The temporal encoder E_t learns temporal dependencies but does not explicitly enforce intrinsic shape features through time, which results in a limited accuracy for the shape identity. To improve this, we define a loss that ensures the preservation of intrinsic properties of the identity shape over all the sequences under consideration. In practice, the intrinsic features are represented by edge lengths on the identity mesh. The underlying assumption is that human shapes undergo near-isometric transformations when moving, which does not impact geodesic distances. Although restrictive, this assumption has proven efficient to enforce shape preserving deformations, as shown in, *e.g.*, [18, 12, 5, 25].

The intrinsic loss writes:

$$\mathcal{L}_{int} = \sum_{t} \sum_{\{(i,j)\}} \left| \frac{||\tilde{v}_t^i - \tilde{v}_t^j||}{||v^i - v^j||} - 1 \right|, \tag{1}$$

where $\{(i,j)\}$ is the set of edges on the template mesh, t the time and v^i, $\tilde{v}_t^i$ vertices on the ground truth and the estimation at t respectively.

3.2.2 Extrinsic Motion Preservation

In order to preserve poses over the deformed shapes, we consider pairwise distances between vertices on meshes. The objective is to preserve the global structure of the shape, such as body contacts and relative distances between limbs.

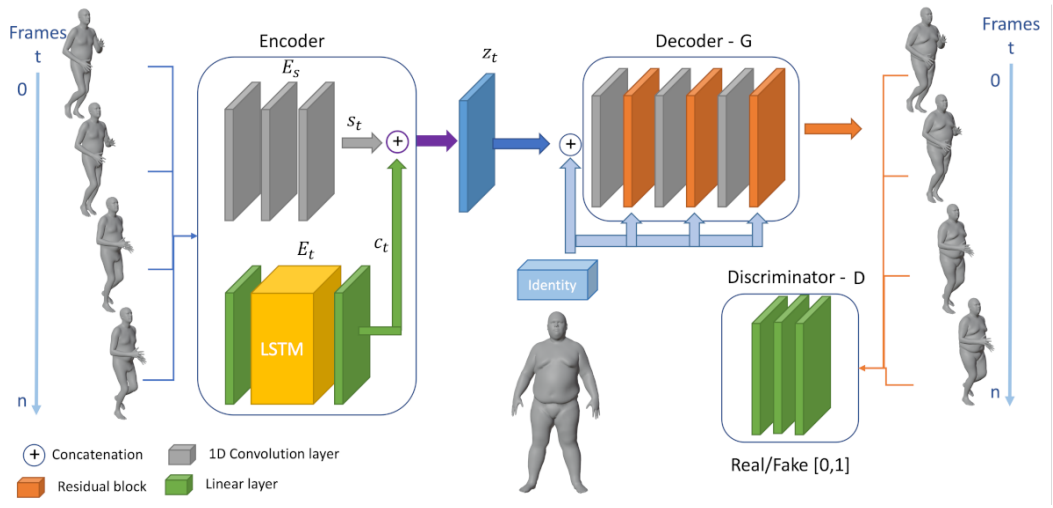

Figure 2. Network architecture: The spatial encoder E_s considers input mesh vertex locations and yields a feature s_t per input frame. The temporal encoder E_t is an LSTM network that takes as input the vertex locations over all frames and outputs a hidden state vector c_t per frame. The decoder G produces the template mesh vertex locations for the new identity frames. For the discriminator D we adopt a GAN framework to predict real or fake mesh shapes. See supplementary materials for detailed network architectures and hyper-parameter specifications.

Distances between nearby vertices on the mesh tend to encode intrinsic shape properties, but distances between farther vertices characterize poses. Both should correspond to the estimated shapes and ground truth shapes. While these distances include the edge length mentioned before, the constraint applies here on a per frame basis, which impacts the back-propagation differently than the intrinsic feature preservation and facilitates the decoder convergence in practice.

Equation 2 below defines the shape pose loss we use. It compares at each time t the distance matrices $\mathcal{U}^{i,j}(.)$ of the generated vertices $\tilde{V}_t$ and of the ground-truth vertices V_t. The distance matrix $\mathcal{U}^{i,j}(.)$ is the upper triangular matrix with all the Euclidean distances between mesh vertices considered pairwise.

$$\mathcal{L}_{ext} = \frac{1}{n}\sum_{t} ||\mathcal{U}^{i,j}(V_t) - \mathcal{U}^{i,j}(\tilde{V}_t)||_F^2, \tag{2}$$

where $||.||_F$ denotes the Frobenius norm.

In addition to the spatial distances between mesh vertices, we could also consider the temporal distances covered by the vertices between successive frames as extrinsic features. However, our experiments did not demonstrate significant improvements using such distances, and we use them for evaluation purposes only.

3.3. Model Learning

The network is trained in an end-to-end fashion, where the encoder, decoder, and discriminator models are learned simultaneously, as illustrated in Figure 2. The overall training objective function $\mathcal{L}_g$ is defined as follows:

$$\mathcal{L}_g = \mathcal{L}_{int} + \mathcal{L}_{ext} + \mathcal{L}_{rec} + \mathcal{L}_{adv}, \tag{3}$$

where $\mathcal{L}_{int}$ and $\mathcal{L}_{ext}$ are the intrinsic and extrinsic feature losses introduced before. $\mathcal{L}_{rec}$ is the reconstruction loss:

$$\mathcal{L}_{rec} = \sum_{\{t\}}\sum_{\{i\}} ||v_i - \tilde{v}_i||^2, \tag{4}$$

that accounts for the L2 norm error between the estimated vertices $\tilde{v}_i$ and the ground truth vertices v_i at each frame. This loss alone does not guarantee realistic surfaces since it does not enforce any spatial consistency between the estimated vertices. To this aim, we introduce the discriminator D that is trained to maximize the probability of predicting the correct label of both training examples and the decoder synthesis. This is implemented as a classical adversarial loss [11]:

$$\begin{aligned}\mathcal{L}_{adv} = \min_G \max_D \mathbb{E}_{V_t \sim p_{data}}[\log(D(V)] \\ + \mathbb{E}_{z_t \sim p_{z_t}}[\log(1 - D(G(z_t)))],\end{aligned} \tag{5}$$

where p_{data} and $p(z_t)$ denotes the distribution of the training set and prior distribution for G. The min and max refer to the minimization of the decoder G loss and the maximization of the discriminator's D loss. The respective contributions of the above different losses are evaluated with an ablation study that is presented in Table 2.

3.4. Implementation Details

All the experiments were carried on a PC with a single NVIDIA Quadro RTX 5000, 16GB. The network is trained on 10^3 epochs where each epoch sees 10^3 examples. It is optimized using the Adam optimizer [16] with a momentum of 0.9, and with a learning rate of 0.001 for both decoder G and discriminator D, where training data are distinct shapes from the testing data. Empirically the weights of the objective function $\mathcal{L}_g$ were found to best perform when all set to 1. The batch size is fixed to one for all settings, and each batch contains 4 3D mesh frames as a list of vertex coordinates. Please consult the supplementary material for more details on training and testing shapes. The encoder E_s receives as input the frames in a batch of size 4, consequently outputting 4 latent variables $s_t = [s_0, ..., s_3]$. On the other hand, the encoder E_t receives as input a sequence of 4 frames with a batch of size 1. This will generate four hidden state variables $c_t = [c_0, ..., c_3]$, which are then concatenated with s_t to form $z_t = [z_0, ..., z_3]$ that is fed into the decoder G.

The training time takes around 41 hours where NPT [25] takes 24 hours. At inference, the proposed method takes less than one second, where the optimization methods [5] requires 20 minutes per frame on average.

4. Experiments

This section presents results and evaluations for the proposed temporal shape transfer network. We use for that purpose the AMASS dataset [21]. SMPL based datasets [20], FAUST [6], Dynamic FAUST[7] and realistic clothed people (3DPW) [24] are also used to evaluate shape transfer generalization capabilities with challenging examples. A comparison against state-of-the-art learning-based and optimization methods is given in Table 1 with quantitative results, demonstrating significant improvement in the shape quality and motion preservation. Qualitative results are shown in Figures 5, 7, 3 and 4, demonstrating the ability to preserve motion and shape for long motion sequences, and in Figure 6 to illustrate challenging realistic shape transfers with, *e.g.*, clothed people. An ablation study is presented in Table 2 and Table 3 to quantitatively evaluate the contribution of each module, input sequence lengths and loss functions relative to reconstruction, shape identity transfer, and motion preservation. Code is available at https://github.com/joaoregateiro/TemporalShapeTransferNetwork.

4.1. Datasets

The AMASS dataset [21] is used for training and testing, where 18 and 9 body shapes are randomly generated to create a unique collection of training and testing shapes, respectively. The generation is performed using the parameterized template model (SMPL) [20], which allows control of pose and body shape. The 21 motion sequences are divided into 10 training and 11 testing sequences, which are generated using the provided motion capture dataset containing approximately 26000 frames for training and testing. This was used as ground-truth data to train our network and evaluate the contribution of temporal features on motion preservation and shape transfer (see the supplementary for more details on the shape and pose diversity in training and testing).

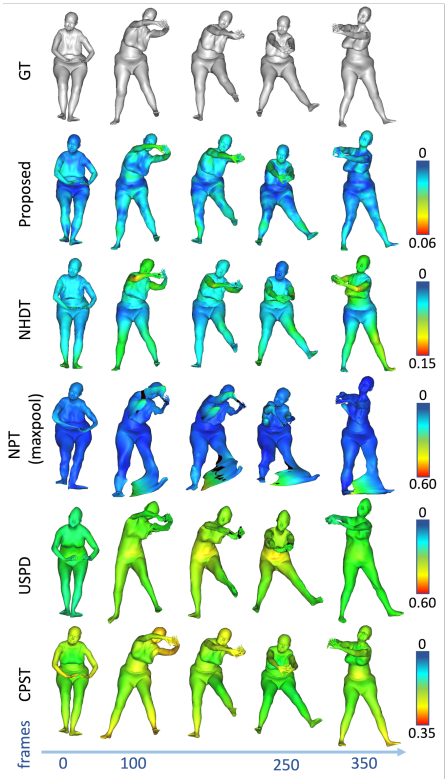

Figure 3. Shape transfer evaluation against state-of-the-art methods. The color coded error is a mean per-vertex distance between predicted and ground-truth (GT) shapes. Note that different normalizations are applied to better visualize performances.

4.2. Quantitative Evaluation

We conducted quantitative comparisons with two metrics: static and temporal shape errors. The static shape error evaluates the reconstruction error using two different metrics. The Root Mean Squared Distance (RMSD) is used to measure the average accuracy with respect to the ground-truth, and the Hausdorff distance gives a good measure of the mutual proximity of two 3D meshes. On the other hand, the temporal shape error measures vertex trajectory error with first order differences over time. This instant velocity error provides additional information on the stability of the

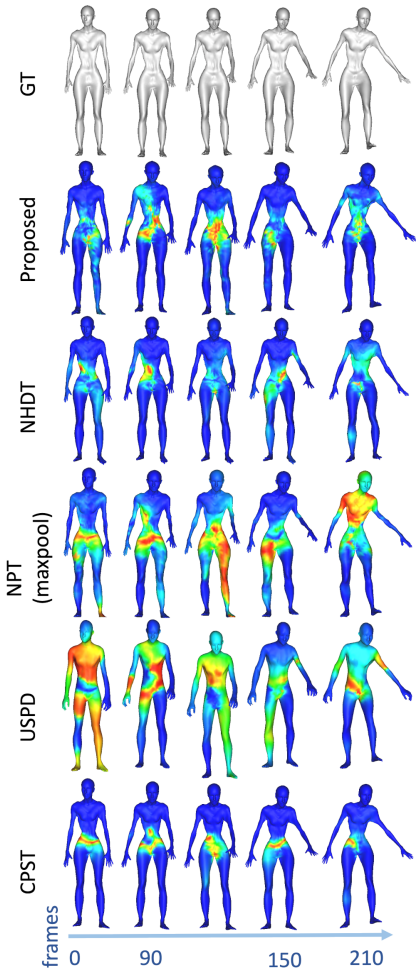

Figure 4. Motion preservation comparison against state-of-the-art methods. The color coded error is a per-vertex first order derivative vector comparison between predicted and ground-truth (GT) shapes. Errors are normalized between 0 and 1.

vertex predictions.

We have compared our method with five state-of-the-art methods: Neural Human Deformation Transfer (NHDT) [4], Neural Pose Transfer (NPT) [25], Unsupervised Shape and Pose Disentanglement (USPD) [28] and Contact Preserving Shape Transfer (CPST) [5] on the same AMASS dataset. These methods are accessible and solve for shape or pose transfer on a single frame basis, which provides informative baseline methods to demonstrate the contribution of temporal features on the problem of human shape transfer.

4.2.1 Static Shape Error

Shape reconstruction errors are evaluated using the RMSD defined as: $RMSD(V,\tilde{V}) = \frac{1}{n}\sqrt{\sum_{i=1}^{n}(v_i - \tilde{v}_i)^2}$, between the generated $\tilde{V}$ and ground truth V mesh vertices $\tilde{v}, v$ respectively. The Hausdorff distance between those 2 meshes is defined as: $d_H(\tilde{V},V) = max\{\sup_{v\in V} d(v,\tilde{V}), \sup_{\tilde{v}\in\tilde{V}} d(\tilde{v},V)\}$, where $d(a,B)$ is the minimal distance from a point a to a set B.

Table 1 illustrates the reconstruction error for training and testing data, demonstrating a significant improvement in shape and pose reconstruction with $0.0108m$ and $0.0283m$ against $0.0547m$ and $0.0367m$ (training and testing with Hausdorff distances) for the best performing model in the state-of-the-art (NPT with maxpooling [25]).

4.2.2 Temporal Shape Error

The temporal shape error is evaluated by comparing velocities with first order differences in vertex locations, *i.e.*, $v_i^{t+1} - v_i^t$ for the vertex i. With these vector values, errors are estimated by considering both the direction with the angle and the L2 norm error between the terminal points of the estimated and ground-truth vectors. For every predicted motion frame and ground-truth, we compute the average means of the angular velocity differences and the L2 norm errors for all vertices and all frames of a sequence. Table 1 shows comparisons with the state-of-the-art for these metrics. Our method performs best overall, however, the CPST [5] optimization approach is the best performing method on the validation set. Although CPST is more sensitive to scaling and body proportions, as illustrated in Figure 3 with higher reconstruction errors, the costly optimization process provides some benefits with more accurate vertex trajectories.

4.3. Qualitative Evaluation

Qualitative results of the proposed network are presented in the Figures 3, and 5 for reconstruction errors, Figure 4 and 5 for velocity errors, Figures 1 and 6 for the ability to generalize over other datasets and Figures 1 and 7 for a sequence example. As illustrated in the latter, results are consistent across the animation and do not introduce noticeable artifacts, such as foot scatting, jitter, or stretching. Figure 3 illustrates the reconstruction error performances of our approach with respect to the state-of-the-art. Figure 4 shows that the proposed method can describe the motion in the regions with significant changes *e.g.* arms and legs, and is comparable to the state-of-the-art. Figure 5 compares the proposed network to the best performing single frame basis state-of-the-art method (NPT maxpool [25]). The errors in Figure 4 are normalized between 0 (blue) and 1 (red). In addition, Figure 6 show that the proposed network can transfer the shape style of unseen realistic characters from realistic body shapes datasets (FAUST and Dynamic FAUST) [6, 7] and realistic clothed people (3DPW) [24], while preserving

Table 1. Comparison of the proposed network against NHDT [4], NPT [25], USPD [28] and CPST [5]. The values represent the average Hausdorff (HDFF) and root mean squared (RMSD) reconstruction errors in meters; the L2 norm errors (L2) and angular errors between velocity vector direction across all motion sequences and for different character shapes. Best results are in bold.

	Proposed		NHDT [4]		NPT maxpool [25]		NPT [25]		USPD [28]		CPST [5]	
Dataset	Train	Val	Train	Val	Train	Val	Train	Val	Train	Val	Train	Val
HDFF(m)	**0.0108**	**0.0283**	0.1902	0.1145	0.0547	0.0357	0.0617	0.0429	0.1769	0.0528	0.0848	0.0799
RMSD(m)	**0.0024**	**0.0072**	0.0734	0.0327	0.0155	0.0092	0.0155	0.0097	0.0704	0.0201	0.0368	0.0328
L2	**0.1136**	0.1713	0.1538	0.2174	0.4488	0.5800	0.6210	0.8506	1.1225	0.8359	0.2533	**0.1549**
direction	**11.30** °	14.80°	15.48°	18.26°	31.05°	36.34°	39.86°	47.47 °	58.10 °	45.79 °	20.26 °	**12.08** °

Table 2. Ablation study of the different losses presented in Section 3. In addition to the metrics in Table 1, the average edge length error (equation 1) over the vertices is given.

	$\mathcal{L}_{rec}$ only		$\mathcal{L}_{rec,adv}$		$\mathcal{L}_{rec,adv,int}$		$\mathcal{L}_{rec,adv,int,ext}$	
Dataset	Train	Val	Train	Val	Train	Val	Train	Val
HDFF(m)	0.0138	0.0384	0.0112	0.0342	0.0129	0.0339	**0.0108**	**0.0283**
RMSD(m)	0.0036	0.0100	0.0026	0.0083	0.0031	0.0085	**0.0024**	**0.0072**
Intrinsic (edge)	0.2347	0.3568	0.2333	0.3606	0.1534	0.2521	**0.1459**	**0.2469**
L2	0.1302	0.2009	0.1312	0.1973	0.1208	0.1816	**0.1136**	**0.1713**
direction	12.59 °	16.50 °	12.36 °	16.39 °	11.86 °	15.26 °	**11.30**°	**14.80** °

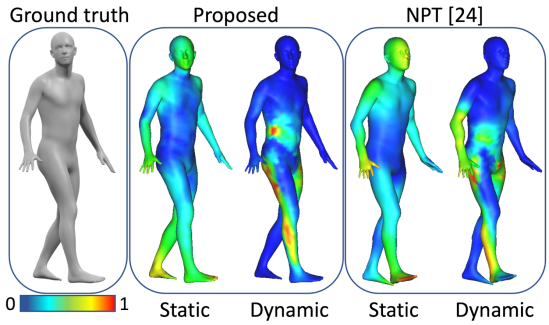

Figure 5. Close-up comparison with state-of-the-art on pose transfer on a single frame basis (NPT [25]). Static: reconstruction errors. Dynamic: velocity errors.

the high-frequency details such as body muscular anatomy and clothing. Note here that the network has no information on clothed shapes at training, illustrating generalization ability. See the supplementary material for more examples of unseen realistic characters and video results on animation sequences.

5. Ablation

An ablation study that evaluates the contribution of the different losses introduced in Section 3 is presented in Table 2. The metric used is detailed in Section 4.2. To further understand the contribution of the intrinsic features on the results, we evaluate the edge length preservation. The comparison is made using the $\mathcal{L}_{int}$ loss (Equation 1) between predicted and ground-truth shapes.

The ablation study shows that the reconstruction $\mathcal{L}_{rec}$ is not sufficient for the task of shape transfer. We see a performance increase in the reconstruction errors with the introduction of the adversarial loss $\mathcal{L}_{adv}$. However, the preservation of the intrinsic shape and motion properties is more or less stable. The latter is clearly improved with the intrinsic shape preservation loss. The addition of the extrinsic loss, hence the entire proposed network, demonstrates the best results on the shape reconstruction errors, identity, and motion preservation.

Table 3 shows the contribution on the number of frames used as input to the temporal encoder E_t in Section 3.1. It is demonstrated that the benefits will incrementally improve with the increase in the number of frames. In this work, we empirically choose four frames.

6. Conclusion

We presented a novel learning-based approach to the problem of motion transfer between digital characters. Our main contribution is to investigate the time dimension for that purpose with a neural network that considers temporal sequences of shape meshes as input instead of single frames as traditional in the state-of-the-art. The temporal aspects are exploited through LSTM networks that encode time dependencies between frames as well as through edge length preservation that enforces intrinsic shape identity consistency over time. Experimental results demonstrate that temporal information contributes to better shape reconstructions in addition to better motion preservations.

7. Acknowledgments

This work has been partially supported by MIAI@Grenoble Alpes (ANR-19-P3IA-0003) and by the European Union's Horizon 2020 research and innovation program under grant agreement No 952.147.

Table 3. Ablation study of the model $\mathcal{L}_{int,ext,rec,adv}$ using different input sequence lengths presented in Section 3.

	1 frame		2 frames		3 frames		4 frames		5 frames	
Dataset	Train	Val	Train	Val	Train	Val	Train	Val	Train	Val
HDFF(m)	0.0161	0.0372	0.0111	0.0411	0.0117	0.0317	**0.0108**	**0.0283**	0.0111	0.0371
RMSD(m)	0.0044	0.0101	0.0026	0.0104	0.0028	0.0084	**0.0024**	**0.0072**	**0.0024**	0.0088
Intrinsic (edge)	0.2171	0.3294	0.2064	0.3284	0.1968	0.3154	**0.1459**	**0.2469**	0.1862	0.3024
L2	0.1475	0.2065	0.1381	0.1884	0.1238	**0.1685**	**0.1136**	0.1713	0.1370	0.2122
direction	12.63 °	17.09 °	12.61 °	15.72 °	11.84 °	**14.76** °	**11.30**°	14.80 °	13.19 °	17.36 °

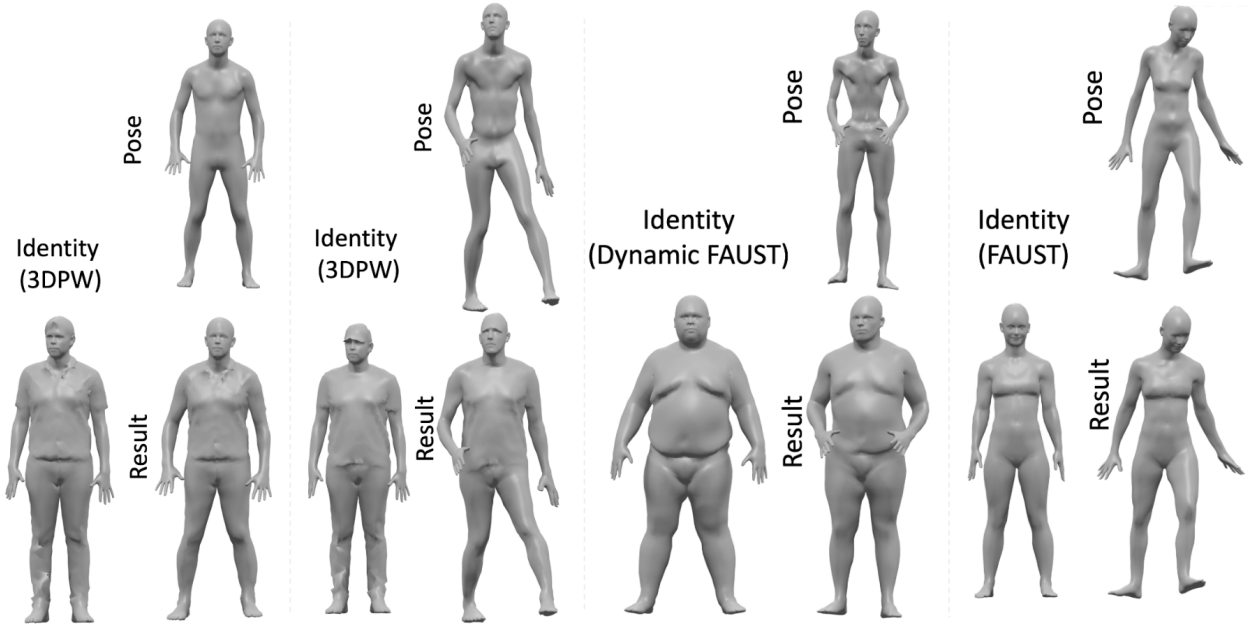

Figure 6. Generalization over real shapes from other datasets (not used for training): 3DW [24], Dynamic Faust [7] and Faust [6], including shapes with clothing as on the top examples.

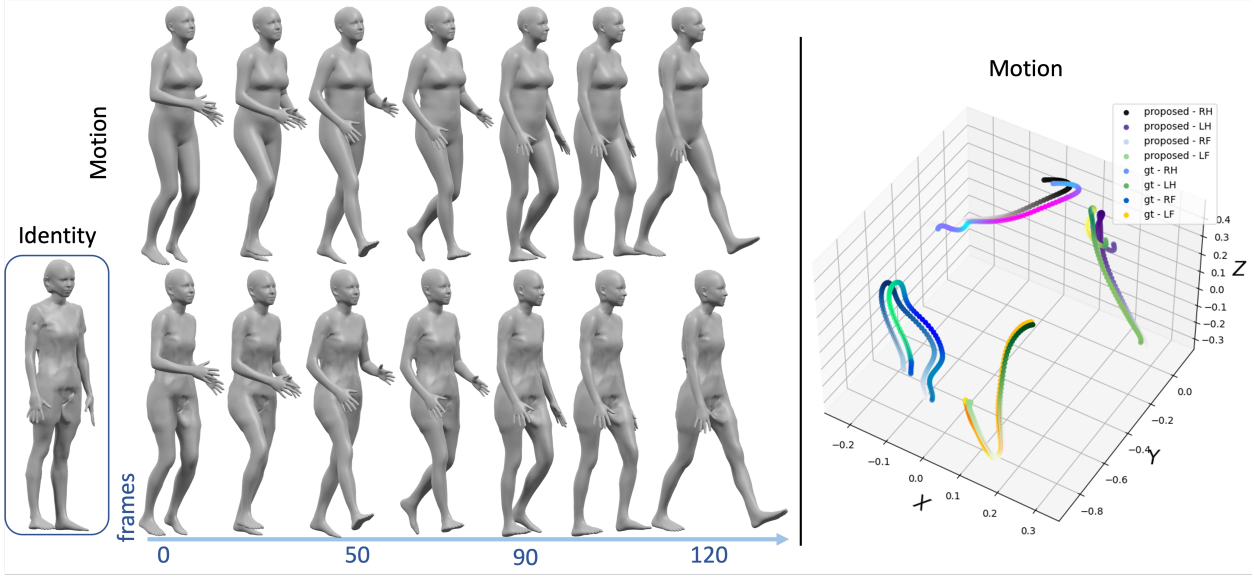

Figure 7. Shape transfer from an unseen realistic shape (identity shape on the left) onto a sequence of 3D shapes (top). On the right, a visualization of both prediction (bottom shapes) and top shape motion for four vertex motions located at the hands and feet, illustrating motion accuracy.

References

[1] Kfir Aberman, Peizhuo Li, Dani Lischinski, Olga Sorkine-Hornung, Daniel Cohen-Or, and Baoquan Chen. Skeleton-aware networks for deep motion retargeting. 2020. 4322

[2] Ilya Baran and Jovan Popović. Automatic rigging and animation of 3d characters. *ACM Trans. Graph.*, 2007. 4321, 4322

[3] Ilya Baran, Daniel Vlasic, Eitan Grinspun, and Jovan Popović. Semantic deformation transfer. 2009. 4321, 4322

[4] Jean Basset, Adnane Boukhayma, Stefanie Wuhrer, Franck Multon, and Edmond Boyer. Neural Human Deformation Transfer. In *International Conference on 3D Vision (3DV)* , 2021. 4321, 4323, 4326, 4327

[5] Jean Basset, Stefanie Wuhrer, Edmond Boyer, and Franck Multon. Contact Preserving Shape Transfer For Rigging-Free Motion Retargeting. In *MIG 2019 - ACM SIGGRAPH Conference Motion Interaction and Games*, 2019. 4322, 4323, 4325, 4326, 4327

[6] Federica Bogo, Javier Romero, Matthew Loper, and Michael J. Black. FAUST: Dataset and evaluation for 3D mesh registration. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2014. 4321, 4325, 4326, 4328

[7] Federica Bogo, Javier Romero, Gerard Pons-Moll, and Michael J. Black. Dynamic FAUST: Registering human bodies in motion. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 4321, 4325, 4326, 4328

[8] Adnane Boukhayma, Jean-Sébastien Franco, and Edmond Boyer. Surface Motion Capture Transfer with Gaussian Process Regression. In *CVPR 2017 - IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 4321, 4322

[9] Haoyu Chen, Hao Tang, Shi Henglin, Wei Peng, Nicu Sebe, and Guoying Zhao. Intrinsic-extrinsic preserved gans for unsupervised 3d pose transfer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 4321, 4322

[10] Lin Gao, Jie Yang, Yi-Ling Qiao, Yu-Kun Lai, Paul L Rosin, Weiwei Xu, and Shihong Xia. Automatic unpaired shape deformation transfer. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2018)*, 2018. 4322

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. 2020. 4324

[12] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan Russell, and Mathieu Aubry. 3D-CODED : 3D Correspondences by Deep Deformation. In *ECCV 2018*, 2018. 4323

[13] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 1997. 4323

[14] Pushkar Joshi, Wen C. Tien, Mathieu Desbrun, and Frederic Pighin. Learning controls for blend shape based realistic facial animation. In *ACM SIGGRAPH 2006 Courses*. Association for Computing Machinery, 2006. 4322

[15] Ladislav Kavan, Steven Collins, Jiří Žára, and Carol O'Sullivan. Geometric skinning with approximate dual quaternion blending. *ACM Trans. Graph.*, 2008. 4321, 4322

[16] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, (ICLR)*, 2014. 4325

[17] Paul G. Kry, Doug L. James, and Dinesh K. Pai. Eigenskin: Real time large deformation character skinning in hardware. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. Association for Computing Machinery, 2002. 4322

[18] Yaron Lipman, Daniel Cohen-Or, Ran Gal, and David Levin. Volume and shape preservation via moving frame manipulation. *ACM Trans. Graphics*, 2007. 4323

[19] Yaron Lipman, Olga Sorkine, David Levin, and Daniel Cohen-Or. Linear rotation-invariant coordinates for meshes. *ACM Trans. Graph.*, 2005. 4321, 4322

[20] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 2015. 4325

[21] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *International Conference on Computer Vision*, 2019. 4321, 4325

[22] Helge Rhodin, James Tompkin, Kwang In Kim, Edilson de Aguiar, Hanspeter Pfister, Hans-Peter Seidel, and Christian Theobalt. Generalizing wave gestures from sparse examples for real-time character control. *ACM Trans. Graph.*, 2015. 4322

[23] Robert W. Sumner and Jovan Popović. Deformation transfer for triangle meshes. In *ACM SIGGRAPH 2004 Papers*. Association for Computing Machinery, 2004. 4321, 4322

[24] Timo von Marcard, Roberto Henschel, Michael Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *European Conference on Computer Vision (ECCV)*, 2018. 4321, 4325, 4326, 4328

[25] Jiashun Wang, Chao Wen, Yanwei Fu, Haitao Lin, Tianyun Zou, Xiangyang Xue, and Yinda Zhang. Neural pose transfer by spatially adaptive instance normalization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 4321, 4322, 4323, 4325, 4326, 4327

[26] Han-Bing Yan, Shi-Min Hu, and Ralph Martin. Skeleton-based shape deformation using simplex transformations. In Tomoyuki Nishita, Qunsheng Peng, and Hans-Peter Seidel, editors, *Advances in Computer Graphics*. Springer Berlin Heidelberg, 2006. 4322

[27] Yizhou Yu, Kun Zhou, Dong Xu, Xiaohan Shi, Hujun Bao, Baining Guo, and Heung-Yeung Shum. Mesh editing with poisson-based gradient field manipulation. *ACM Trans. Graph.*, 2004. 4322

[28] Keyang Zhou, Bharat Lal Bhatnagar, and Gerard Pons-Moll. Unsupervised shape and pose disentanglement for 3d meshes. In *The European Conference on Computer Vision (ECCV)*, 2020. 4321, 4322, 4323, 4326, 4327

[29] Kun Zhou, Weiwei Xu, Yiying Tong, and Mathieu Desbrun. Deformation Transfer to Multi-Component Objects. *Computer Graphics Forum*, 2010. 4322