Seminar Report
NPMs: Neural Parametric Models for 3D Deformable Shapes

Bertan Karacora
bertan.karacora@uni-bonn.de
Matr. Nr.: 3226384

August 31, 2023

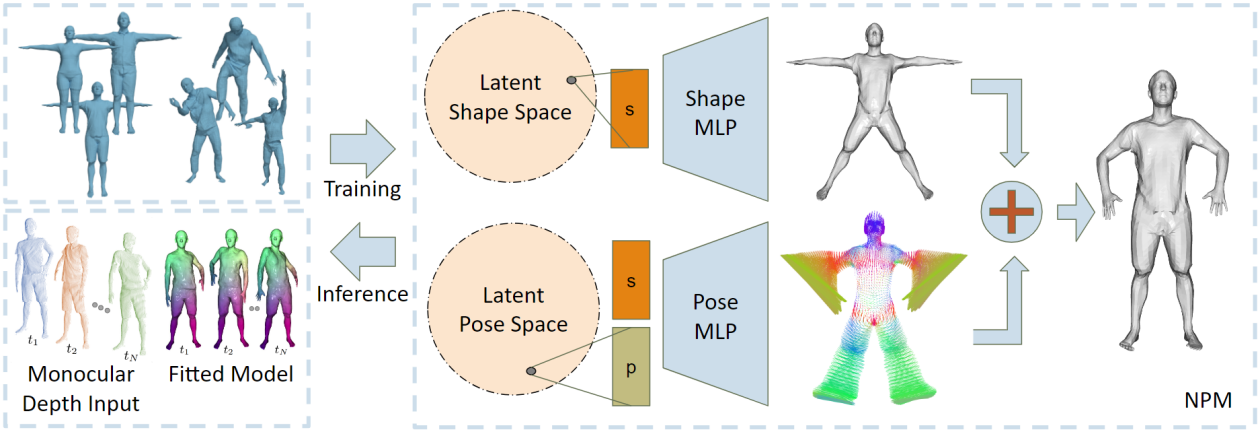

Figure 1: Neural Parametric Models (NPMs) do not require a handcrafted model formulation, instead, they learn a parametrization of 3D deformable shapes from data. NPMs use implicit representations to account for arbitrary topologies, domains, and a high level of detail, decoding surface geometry and pose-induced deformations from two separate latent spaces. The well-regularized nature of these spaces allows for the accurate fitting to new observations in test time.

Abstract

Expressive models are fundamental for reconstructing and tracking of 3D deformable shapes such as human bodies, faces, and hands. Based on their explicit construction and controllable behavior, traditional parametric models have become a popular tool in many visual applications. However, creating a suitable parametrization typically requires a careful and labor-intensive design process. Additionally, the expressiveness of these models is limited, especially with regard to fine-scale details such as clothing and hair. This report deals with Neural Parametric Models (NPMs), as proposed by Palafox, Božič, et al. (2021), which employ learned representations instead, disentangling the dynamics of deformable objects into latent spaces of shape and pose. Leveraging continuous implicit functions that decode shape and pose parameters into a deformed 3D shape, NPMs do not depend on resolution or domain-specific knowledge. In their work, Palafox, Božič, et al. (2021) demonstrate the capabilities of NPMs for the task of fitting unseen deforming shapes given as sequences of monocular depth maps. Furthermore, they give examples of latent-space interpolation as well as shape and pose transfer using NPMs. This report aims to summarize their approach, methodology, and experiments while adding own explanations and critical thoughts. In particular, NPMs show promising potential for constructing accurate models for any class of non-rigid shapes, however, the learned representations prohibit a direct interpretation or a semantically meaningful manipulation of the learned model parameters.

1 Introduction

Accurately modeling 3D deformable shapes such as human bodies, faces, and hands is a crucial and challenging problem in various fields of visual computing. Whereas rigid bodies have an unambiguous geometry (up to translation and rotation), a deformable object may change its physical form significantly and non-uniformly depending on its pose. Applications range from 4D reconstruction, visual

content creation, and avatar design, to virtual try-on and computer-assisted coaching (Tian et al., 2022). To allow for a realistic manipulation of those shapes, it is critical to account for non-rigid deformations when creating models to fit to real-world data.

Drawing ongrowing attention in recent years, different parametric models have been established for the representation of non-rigid shapes, aiming to control distinct shape properties via a set of low-dimensional parameters. Although their construction may be guided by statistical techniques like PCA, the parametrization needs to be specified by hand. For example, in the case of the widely used SMPL model (Loper et al., 2015), this involves the manual creation of a template mesh and a skeleton of rigidly transformable parts to apply blend skinning. SMPL can then be controlled using blend weights. As a consequence, these models are only applicable to a single object domain and restricted by the input information considered in their parametric formulation. The development process is often time-consuming and requires deep expert knowledge. Despite great effort and steady advances, these models lack the expressiveness to cover all aspects observable in deformable shapes, especially regarding dynamically changing details such as clothing and hair (Palafox, Božič, et al., 2021).

Neural Parametric Models, as proposed by Palafox, Božič, et al. (2021), offer an alternative approach for modeling 3D deformable shapes. Instead of defining semantic parameters, they use fully learned parametric spaces that are interpreted by implicit functions. Specifically, a deformable shape is characterized by latent codes of shape and pose. A Neural Parametric Model (NPM) can be learned for any class of objects that has been observed by the model during training, without manual specification of the properties of a target domain, e.g., a kinematic chain, and constraints of connectivity and topology. Afterward, the model can be fit to new observations at test time.

For the construction of 3D deforming shapes from latent codes, i.e., points of the latent spaces, NPMs take advantage of disentangling shape and pose from each other while using the expressiveness of continuous implicit functions. To this end, NPMs incorporate two MLPs trained in auto-decoder fashion. One learns the SDF that decodes a shape code into implicit geometry in some canonical pose. The other learns to predict the non-rigid deviations from the canonically-posed shape. Notably, the continuous implicit representations support arbitrary topologies as well as the accurate description of fine-scale details.

Palafox, Božič, et al. (2021) evaluate the accuracy of NPMs for modeling human bodies with clothing and for modeling hands both in a qualitative and a quantitative fashion. Making use of suitable optimization routines, their approach achieves at-the-time state-of-the-art performances in model fitting to moving clothed humans, i.e., reconstructing and tracking deforming 3D shapes, given as sequences of monocular depth maps. Additionally, they highlight the well-regularized nature of the learned latent spaces of shape and pose showcasing examples of latent space interpolation, shape transfer, and pose transfer.

This report aims to give a condensed description of the work of Palafox, Božič, et al. (2021). In order to do so, the following elaborates on the essential ideas, methods, and experimental results, supplemented with own explanations and novel considerations. Additionally, further examples of related work are taken into account including some more recent developments.

2 Related Work

Traditional parametric models. Typically, to a certain extend, the input parameters of traditional models describe changes of the shape in a semantically meaningful way. Considering human bodies, these often disentangle shape and pose following the idea of SCAPE (Anguelov et al., 2005). Notably, the very popular Skinned Multi-Person Linear model (SMPL; Loper et al., 2015) is based on blend skinning, assuming a skeleton, a template mesh with per-vertex blend weights and a predefined set of rotatable joints. SMPL constructs a human body from parameters including the coefficients of principal shape components and the weights of pose blend shapes. The latter may be computed from explicit joint rotation angles.

Generally, many modern parametric models incorporate shape priors extracted by statistical analysis of training samples into explicit reconstruction functions (Tretschk et al., 2023). However, these are highly domain-specific, require great effort to design, and need to resort to extensions to include further details. For SMPL, there is a long list of such extensions, e.g., focusing on more expressive faces and hands (Pavlakos et al., 2019), making pose deformations shape-dependent (Osman et al., 2020), or using vertex displacements to consider clothing (SMPL+D; Alldieck et al., 2019). In consideration of these efforts, the advantages of a versatile learned model applicable to any object domain with all the observable details seem apparent.

Neural implicit representations. Instead of raw explicit representations such as voxels, meshes, and point clouds, implicit functions aim to capture geometry as a continuous field, e.g., as the signed distance to the object's surface. Importantly, this lifts the constraint of resolution which comes with an extensive increase in memory and computational cost, for example, if one would try to apply convolutional neural networks. Xie et al. (2022) provide a useful overview of both methods and applications concerning neural fields.

In many cases, for learning an implicit function, an auto-

encoder architecture is employed to enable end-to-end supervision. In contrast to this, DeepSDF (Park et al., 2019) motivates the auto-decoder approach, omitting the encoder part by directly learning the latent space jointly with the decoder network weights. DeepSDF does not disentangle shape and pose which results in uncorrelated latent codes given different poses of the same deformable shape. In more recent work, Mu et al. (2021) extend DeepSDF to incorporate both a shape code and an articulation code, similar to NPMs (Palafox, Božič, et al., 2021). However, the latter is represented explicitly as joint angles and cannot capture shape-dependent pose deformations, e.g., of soft materials. Further ideas introduced by Wei et al. (2022) focus on learning appearance rather than the inherent pose but do not require surface correspondences. Leveraging the versatile representation power of implicit functions, all of the above may be applied to any object domain other than human bodies as well.

Fitting neural models to non-rigid objects. For fitting to general deformable shapes there is a large number of proposed learning-based models that exceeds the limitations of this report. In the original work of Palafox, Božič, et al. (2021), two approaches mentioned in particular are OFlow (Niemeyer et al., 2019) and IPNet (Bhatnagar et al., 2020). OFlow represents 4D dynamics as an implicit function mapping a motion vector to every point in space and time, so fitting the model results in a continuous 4D reconstruction but is limited to very short motion sequences. IP-Net Bhatnagar et al., 2020 learns an implicit function that classifies body, clothing, and background. Then it predicts correspondences to SMPL which enables the reconstruction of a completely controllable SMPL+D registration during inference.

More recently, SPAMs (Palafox, Sarafianos, et al., 2022) were proposed as a direct extension of NPMs. SPAMs incorporate a semantic structure and learn a multitude of local NPMs, each specialized on one distinct part of the deformable shape. It is shown that SPAMs outperform NPMs when decomposing a human body into body segments such as arms and legs. Giebenhain et al. (2023) observe similar improvements using structured NPMs focusing on modeling human heads and faces. gDNA (Chen et al., 2022) is a generative skinning-based model that creates a more accurate fitting to unseen shapes but is limited to clothed human bodies. H4D (Jiang et al., 2022) controls an underlying SMPL model with motion and auxiliary latent codes, outperforming NPMs on depth completion, i.e., reconstruction from monocular depth. Geometric-Aware NPMs (Mohamed and Agapito, 2022) learn local features of geometric structure and enforce local regularity improving over NPMs in fitting to complete 3D scans.

3 Method

This section provides explanations of the methods of Palafox, Božič, et al. (2021) for the construction of NPMs. Essentially, this can be structured into 4 sections: First, the overall composition of NPMs from disentangled implicit functions that decode latent codes. Next, the training process with regards to the latent spaces of shape and pose, respectively. Finally, the test-time optimization process that enables learned NPMs to fit to new observations.

3.1 Neural Parametric Models

NPMs are a learned alternative to traditional parametric models for 3D deformable shapes. Instead of designing a model explicitly, NPMs are learned from data without the need for any domain-specific constraints, manual annotations, or object templates. The key idea is to learn both a suitable parametrization, i.e., the spaces of possible shapes and poses, and the construction functions jointly in auto-decoder fashion. Accordingly, the training process of NPMs closely follows that of DeepSDF (Park et al., 2019). To consider non-rigid deformations, the approach aims to disentangle shape and pose parameters from each other. As points in latent spaces, i.e., as latent codes, these can be decoded into implicit representations of the geometry and the per-point surface deformations. An NPM can be fit to dynamic 4D data by reconstructing a single suitable shape code and an according set of pose codes.

In general, learning latent encodings can be performed with an auto-encoder. Composed of two neural networks, the encoder first generates a latent representation that is eventually decoded by a decoder network to obtain a reconstruction of the input. In contrast, an auto-decoder is composed solely of the decoder part. It learns the latent space jointly with the network weights during training, thus, omitting the encoder. As a consequence, auto-decoders may still be trained in a sort of self-supervision setting but fitting to unseen observations mostly builds upon test-time inference instead of forward encoding. Specifically, NPMs are composed of two multilayer perceptrons (MLPs) that decode the posed shape into an implicit representation. Given the corresponding latent code, one is estimating the signed distance function that describes the canonically posed shape's surface as zero-isosurface. The other is predicting the flow vectors from the canonical to the deformed pose.

For training, NPMs require a dataset containing the meshes of differently posed identities. Importantly, for each identity of this dataset, there needs to exist one sample shape in a predefined canonical pose, and at least a certain number of other samples in various other poses, each with per-identity dense surface registrations to the canonical pose. As emphasized by Palafox, Božič, et al. (2021), these requirements are

fulfilled by many datasets, e.g. CAPE (Ma et al., 2020), obtained from 3D scans of human bodies in arbitrary clothing, since no registrations between the identities are needed. This report covers the most important ideas and mathematical formulations. For details on the implementation, the reader may refer to the original publication.

3.2 Shape Space

The shape space is the latent space of shape vectors or codes, representing all possible canonically posed shapes of a certain class or domain. It is determined by the learned shape function f_{θ_s}. Conditioned on the shape code $\boldsymbol{s}_i$ of identity i, this function becomes the signed distance function (SDF) which maps any 3D position $\boldsymbol{x}$ to the orthogonal distance $\tilde{d}$ from that position to the shape's surface:

$$\begin{aligned} f_{\theta_\mathrm{s}} : \mathbb{R}^{D_\mathrm{s}} \times \mathbb{R}^3 &\to \mathbb{R}, \\ (\boldsymbol{s}_i, \boldsymbol{x}) &\mapsto f_{\theta_\mathrm{s}}\left(\boldsymbol{s}_i, \boldsymbol{x}\right) = \tilde{d} \end{aligned} \tag{1}$$

The sign classifies whether $\boldsymbol{x}$ is encapsulated by the shape's surface or not. Consequently, an explicit representation as a mesh can be recovered from the SDF as its zero-isosurface by applying standard techniques such as Marching Cubes (Lorensen and Cline, 1987). As a side effect, surface normals can be estimated as the inverse gradient direction of the SDF. For human bodies, a usual choice of the canonical shape is the T-pose.

The function f_{θ_s} is learned by an MLP from a dataset of S shape identities, only considering their canonical poses. For each identity i, a total of N_s points $\left\{\boldsymbol{x}_i^k\right\}_{k=1}^{N_\mathrm{s}} \in \mathbb{R}^3$ is sampled and their corresponding ground truth SDF values $\left\{d_i^k\right\}_{k=1}^{N_\mathrm{s}} \in \mathbb{R}$ are computed. This requires a normalization of all meshes to the unit bounding box determined by the largest bounding box in the dataset. To obtain the correct sign of the distances, the canonically posed shapes are made watertight. Subsequently, the bounding box is sampled, a small amount uniformly and a larger amount close to the shape's surface. In the end, these samples are combined. The resulting samples are used to formulate the reconstruction energy enabling a gradient-based optimization.

A derivation of this reconstruction energy is not given by Palafox, Božič, et al. (2021) but to explain the foundational concepts in this report, it is inferred from Park et al. (2019). In probabilistic terms, the posterior over shape code $\boldsymbol{s}_i$ can be formulated as follows:

$$p_{\theta_s}\left(\boldsymbol{s}_i \middle| \boldsymbol{x}_i^k, d_i^k\right) = p\left(\boldsymbol{s}_i\right) \prod_{k=1}^{N_\mathrm{s}} p_{\theta_s}\left(d_i^k \middle| \boldsymbol{s}_i; \boldsymbol{x}_i^k\right) \tag{2}$$

The SDF likelihood is assumed to be of the form:

$$p_{\theta_s}\left(d_i^k \middle| \boldsymbol{s}_i; \boldsymbol{x}_i^k\right) = \exp\left(-\mathcal{L}_\mathrm{s}\left(f_{\theta_s}\left(\boldsymbol{s}_i, \boldsymbol{x}_j\right), d_i^k\right)\right) \tag{3}$$

where $\mathcal{L}_\mathrm{s}$ is a truncated ℓ_1-loss on the predicted SDF $\tilde{d}_i^k$:

$$\mathcal{L}_\mathrm{s}(\tilde{d}_i^k, d_i^k) = \left| \tilde{d}_i^k\big|_{[-\delta,\delta]} - d_i^k\big|_{[-\delta,\delta]} \right|, \tag{4}$$

$$d|_{[-\delta,\delta]} = \min\left(\max\left(-\delta, d\right), \delta\right) \tag{5}$$

The clamping accounts for a focus on surface details at the expense of slower ray-casting due to unreliable step size estimates outside this interval.

Furthermore, as a prior, the distribution over the latent codes is assumed to be a Gaussian: $p(\boldsymbol{s}_i) \sim \mathcal{N}\left(\boldsymbol{0}, \sigma^2 \boldsymbol{I}\right)$. Now, after applying the logarithm to Eq. 2, the objective function can be obtained as the maximum a posteriori. The shape space is learned by minimizing the negative, considering all S identities:

$$\underset{\theta_\mathrm{s}, \{\boldsymbol{s}_i\}_{i=1}^S}{\arg\min} \sum_{i=1}^{S} \left(\sum_{k=1}^{N_\mathrm{s}} \mathcal{L}_\mathrm{s}(f_{\theta_\mathrm{s}}(\boldsymbol{s}_i, \boldsymbol{x}_i^k), d_i^k) + \frac{\|\boldsymbol{s}_i\|_2^2}{\sigma_\mathrm{s}^2} \right) \tag{6}$$

Notably, the latent codes in the train set $\{\boldsymbol{s}_i\}_{i=1}^S$ and the neural weights θ_s of the shape MLP are optimized jointly.

3.3 Pose Space

The pose space is the latent space of pose codes, representing the possible poses of the shapes from the shape space. It is determined by the learned pose function f_{θ_p}. Conditioned on the shape code $\boldsymbol{s}_i$ of identity i and the pose code $\boldsymbol{p}_j$ of pose instance j, this function predicts the flow field which maps any 3D position $\boldsymbol{x}$ to the deformation vector $\Delta\tilde{\boldsymbol{x}}$, i.e., the offset that moves point $\boldsymbol{x}$ from its location in the canonical pose to the corresponding location in the j-th pose:

$$\begin{aligned} f_{\theta_\mathrm{p}} : \mathbb{R}^{D_\mathrm{s}} \times \mathbb{R}^{D_\mathrm{p}} \times \mathbb{R}^3 &\to \mathbb{R}^3, \\ (\boldsymbol{s}_i, \boldsymbol{p}_j, \boldsymbol{x}) &\mapsto f_{\theta_\mathrm{p}}(\boldsymbol{s}_i, \boldsymbol{p}_j, \boldsymbol{x}) = \Delta\tilde{\boldsymbol{x}} \end{aligned} \tag{7}$$

The pose function yields meaningful values only close to the canonically posed shape's surface. To disentangle shape and pose effectively, the function is conditional on the shape code as well, since the pose-dependent flow is not the same for different shapes. To illustrate, consider a surface point on the a foot of a large and a small person and what deformation offsets correspond to the persons lifting the leg.

Learning the pose space requires a set of P posed shapes, each showing a pose j of an identity i, with dense surface correspondences between its canonical pose and the present pose j. First, a total of N_p surface points is sampled from the mesh of each i-th canonically-posed identity in the dataset, along with the barycentric weights with regard to the triangle it is located on, and the triangle's normal. Now, adding a small random displacement $\delta\boldsymbol{n}$ leads to the sample points $\left\{\boldsymbol{x}_i^k\right\}_{k=1}^{N_\mathrm{p}} \in \mathbb{R}^3$ near the shape's surface. Using the same barycentric weights and displacement vectors, each j-th

posed shape available for the identity is sampled in the same way yielding corresponding points $\left\{\boldsymbol{x}_j^k\right\}_{k=1}^{N_\mathrm{p}} \in \mathbb{R}^3$. Finally, the deformation is sampled as the difference vector $\Delta\boldsymbol{x}_{ij}^k = \boldsymbol{x}_j^k - \boldsymbol{x}_i^k$.

With analogous derivations, the objective function is formulated as follows:

$$\underset{\theta_\mathrm{p}, \{\boldsymbol{p}_j\}_{j=1}^P}{\arg\min} \sum_{\substack{j=1 \\ i=m[j]}}^{P} \left(\sum_{k=1}^{N_\mathrm{p}} \mathcal{L}_\mathrm{p}(f_{\theta_\mathrm{p}}(\boldsymbol{s}_i, \boldsymbol{p}_j, \boldsymbol{x}_i^k), \Delta\boldsymbol{x}_{ij}^k) + \frac{\left\|\boldsymbol{p}_j\right\|_2^2}{\sigma_\mathrm{p}^2} \right) \tag{8}$$

where $m[\cdot]$ is a dictionary that maps the index j of a posed shape to the corresponding index i of its canonical shape. In this case, the latent pose codes in the train set $\{\boldsymbol{p}_j\}_{j=1}^P$ and the neural weights θ_p of the pose function are optimized at the same time, whereas the shape codes $\{\boldsymbol{s}_i\}_{i=1}^S$ are considered to be fixed to enable the disentanglement of shape and pose.

3.4 Test-time Optimization

The learned latent spaces of shape and pose can be searched at test time to fit an NPM to a 3D deformable shape. In particular, the well-regularized nature of the learned spaces allows for a direct gradient-based optimization on these spaces, similar to the energy minimization during training. Palafox, Božič, et al. (2021) demonstrate this idea for the task of fitting a model to a moving person, given as a video with a length of a few seconds. Rather than RGB values, the presented process assumes that the target observations include spatial 3D information, at least partially. Therefore, they consider a sequence of L monocular depth maps. The test-time optimization aims to solve for the single latent shape code $\tilde{\boldsymbol{s}}$ and the L per-frame latent pose codes $\{\tilde{\boldsymbol{p}}_j\}_{j=1}^L$ that best explain the observations.

As a preprocessing step, the depth maps are projected into a discretized 3D grid which can be considered a partial SDF field. The SDF values of the visible volume can be estimated only from the depth, whereas everything behind the shape's boundary is unknown. Therefore, a volumetric mask M_o is created to indicate the occluded regions. In the loss computation, only a small boundary region of the shape's interior, i.e., a 0.01 wide margin is considered valid (in normalized units).

As the NPM only incorporates two decoder MLPs, Palafox, Božič, et al. (2021) recommend training two additional encoders on the learned latent codes of the training dataset in order to get a suitable initialization of $\tilde{\boldsymbol{s}}$ and $\{\tilde{\boldsymbol{p}}_j\}_{j=1}^L$. However, this is a non-essential step and a random initialization of the latent codes is also possible. For sampling surface points, the estimated shape (in canonical pose) is constructed using its initial shape code. Then, N_t points are sampled from its surface. The final sample points $\{\boldsymbol{x}_k\}_{k=1}^{N_\mathrm{t}}$ are obtained by adding random displacements.

Having these samples, model fitting is performed using a similar reconstruction loss as before but with a few regularization terms added:

$$\underset{\boldsymbol{s}, \{\boldsymbol{p}_j\}_{j=1}^L}{\arg\min} \sum_{j=1}^{L} \sum_{\boldsymbol{x}_k} \mathcal{L}_\mathrm{r} + \mathcal{L}_\mathrm{c} + \mathcal{L}_\mathrm{t} + \mathcal{L}_\mathrm{icp} \tag{9}$$

The reconstruction loss is defined as

$$\mathcal{L}_\mathrm{r} = M_\mathrm{o}\mathcal{L}_\mathrm{s}\left(f_{\theta_s}(\boldsymbol{s}, \boldsymbol{x}_k), \left[\boldsymbol{x}_k + f_{\theta_\mathrm{p}}(\boldsymbol{s}, \boldsymbol{p}_j, \boldsymbol{x}_k)\right]_\mathrm{sdf}\right) \tag{10}$$

where $[\cdot]_\mathrm{sdf}$ is a trilinear interpolation of the SDF grid and $\mathcal{L}_\mathrm{s}$ is the truncated loss from Eq. 4. Another term is derived from the assumption of a Gaussian prior distribution, the same as during training:

$$\mathcal{L}_\mathrm{c} = \frac{1}{\sigma_\mathrm{s}^2}\left\|s\right\|_2^2 + \frac{1}{\sigma_\mathrm{p}^2}\left\|p_j\right\|_2^2 \tag{11}$$

Moreover, a temporal consistency term is introduced, enforcing similarity between the current frame j and its neighboring frames $Q = \{j-1, j+1\}$, controlled via λ_t:

$$\mathcal{L}_\mathrm{t} = \lambda_\mathrm{t} \sum_{q \in Q} \left\|f_{\theta_\mathrm{p}}(\boldsymbol{s}, \boldsymbol{p}_j, \boldsymbol{x}_k) - f_{\theta_\mathrm{p}}(\boldsymbol{s}, \boldsymbol{p}_q, \boldsymbol{x}_k)\right\|_2^2 \tag{12}$$

Finally, to make the fitting more robust, the last term is based on the idea of Iterative Closest Point (ICP), controlled via λ_icp. Let $\mathcal{R}$ be a set of predicted points within an extremely small margin around the j-th deformed shape's surface. Then for each point q extracted from the input depth map Q_j, there should exist a nearest neighbor $\mathrm{NN}(q)$ in the predictions $\mathcal{R}$ that is located closely:

$$\mathcal{L}_\mathrm{icp} = \lambda_\mathrm{icp} \sum_{q \in Q_j} \left\|q - \mathrm{NN}_\mathcal{R}(q)\right\|_2 \tag{13}$$

After having estimated suitable parameters, the entire sequence can be reconstructed by sampling the canonical pose from the shape MLP once and the deformations from the pose MLP on a per-vertex and per-frame basis.

4 Experiments

Palafox, Božič, et al. (2021) compare NPMs against the state-of-the-art (at the time) for fitting to monocular depth sequences of human bodies. Specifically, they evaluate SMPL (Loper et al., 2015) with an initial estimate gained from OpenPose (Cao et al., 2019), OFlow (Niemeyer et al., 2019), and IPNet (Bhatnagar et al., 2020). Furthermore, they explore latent space interpolation, shape transfer, and pose transfer. They also demonstrate the flexibility of NPMs by training an NPM for hands. However, to illustrate their main results in this report, it is sufficient to focus on human body fitting.

4.1 Model Fitting

First, an NPM instance is trained on clothed human bodies. The training data combines selected identities from CAPE (Ma et al., 2020), AMASS (Mahmood et al., 2019), and the synthetic dataset DeformingThings4D (Li et al., 2021). To evaluate NPM fitting, a test set is generated from a few unseen sequences of CAPE. These consist of real-world 3D scans of clothed humans performing certain motion sequences. Each scan sequence includes a corresponding SMPL+D registration, i.e., the parameters of shape and the poses to create an SMPL model of the person with added free deformations to account for the clothing (which SMPL does not by itself). Importantly, these deformations are non-parametric raw offsets of the SMPL mesh which need to be registered *per frame*. This is a crucial observation because this motivates why SMPL+D is not a suitable model of clothed humans, although it provides the ground truth registrations in these experiments.

The experimental results are assessed from both a qualitative and quantitative perspective. For the latter, the reconstruction accuracy is averaged over all time steps using three different metrics explained in the following. Volumetric Intersection over union (IoU) quantifies the spatial overlap between the predicted deformed shape and the ground truth SMPL+D mesh. This is straightforward as it only involves deciding whether or not each random sample is inside the shape's surface in both the prediction and the ground truth, and counting the portion where both coincide. The Chamfer-ℓ_2 distance (C-ℓ_2) measures how well the ground truth could be mapped as a template to the predicted mesh. It accumulates the ℓ_2 norm between all template points and its nearest neighbor in the prediction. Therefore, a low Chamfer distance can be interpreted as both accurate and complete reconstruction in a rigid sense. Lastly, the End-Point Error (EPE) measures the deviation of the predicted motion from the ground truth deformation flow (also using the Euclidean norm) based on regular keyframes.

Tab. 1 shows quantitative results for NPM fitting to test data from CAPE. The qualitative performance is illustrated in Fig. 2. Considering comparable methods, NPMs show better capabilities to reconstruct and track the deforming shape, combining well-regularized optimization over the latent spaces with a high level of detail offered by the implicit representations.

Palafox, Božič, et al. (2021) decided to compare their experimental results against OFlow (Niemeyer et al., 2019) and IP-Net (Bhatnagar et al., 2020) which they considered state-of-the-art in fitting to monocular depth sequences at the time. Both of these are based on deep learning and implicit representations. Additionally, they compare with the well-known SMPL model (Loper et al., 2015) which is based on template skeleton and linear modulation of parametric basis vectors, i.e., blend shapes extracted with PCA. To further guide the SMPL fitting, they gain predictions of the joint locations in the observed shape from OpenPose (Cao et al., 2019). Notably, OFlow can only model very short sequences of up to 17 frames, so specially adapted train and test datasets have been arranged.

Model	IoU	C-ℓ_2 ($\cdot 10^{-3}$)	EPE ($\cdot 10^{-2}$)
OpenPose+SMPL	0.68	0.243	2.82
OFlow	0.55	0.755	2.65
IP-Net	0.82	0.034	2.52
NPMs	0.83	0.022	0.74

Table 1: Quantitative results and comparison with state-of-the-art (at the time) models for non-rigid 4D reconstruction from monocular depth. Note that OFlow is evaluated on shorter sequences due to its limitations. Adapted from Palafox, Božič, et al. (2021).

Palafox, Božič, et al. (2021) do similar tests on some synthetic identities from DeformingThings4D. Furthermore, using the flexibility of NPMs, they also capture the performance on a dataset of human hands, specifically, MANO (Romero et al., 2017). In these experiments, NPMs show a similar level of accuracy and detail without any significant new insights different from those observable in the human body fitting experiments.

4.2 Interpolation and Transfer

In the learned spaces of shape and pose, a set of known examples can be interpolated to obtain any linear combination of either their shapes, their poses, or both. As can be seen in Fig. 3, the spaces are well-regularized and allow for a smooth traversal.

As an edge case of interpolation, NPMs can be utilized to perform shape and pose transfer by simply copying the according parametric code from one example to another. This allows both replacing the identity of a deforming shape with another identity and re-posing a target shape with the foreign poses. In other words, one might capture an arbitrary motion once and apply it to any shape registered to the model, artificially re-animating it with realistic deformations.

5 Discussion

While NPMs show great potential for conveniently constructing parametric models and fitting them to real-world data, several limitations remain. This section is meant to discuss the strengths and weaknesses of NPMs while focusing on own perspectives that have not been mentioned in the original work of Palafox, Božič, et al. (2021).

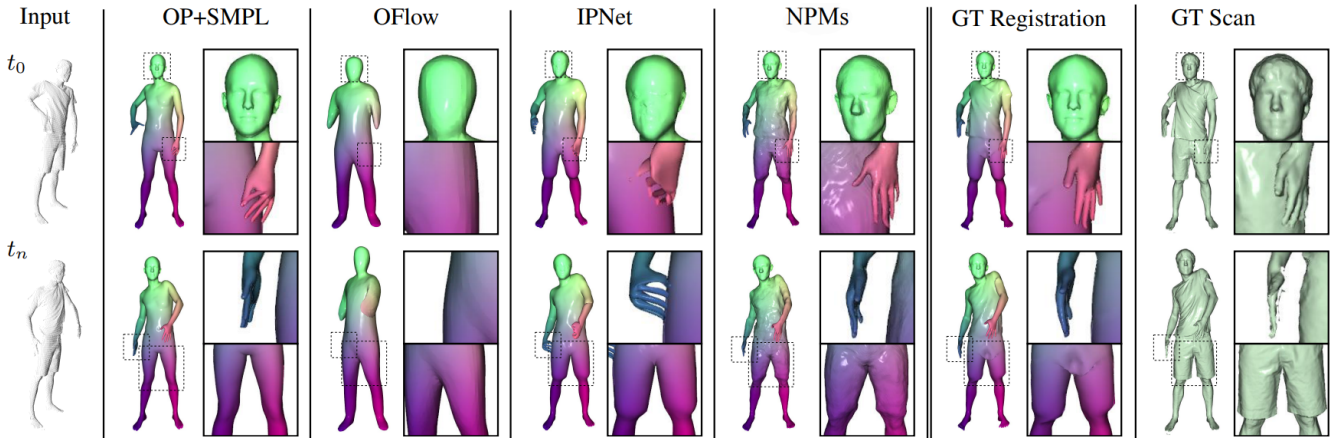

Figure 2: Qualitative results and comparison with state-of-the-art (at the time) models for non-rigid 4D reconstruction from monocular depth. Given a sequence of depth maps (left), each model estimates the best-fit parametric representation and reconstructs the deforming shape. The last two columns show the ground truth SMPL+D registration and the original scans from which the input depth maps are rendered. Adapted from Palafox, Božič, et al. (2021).

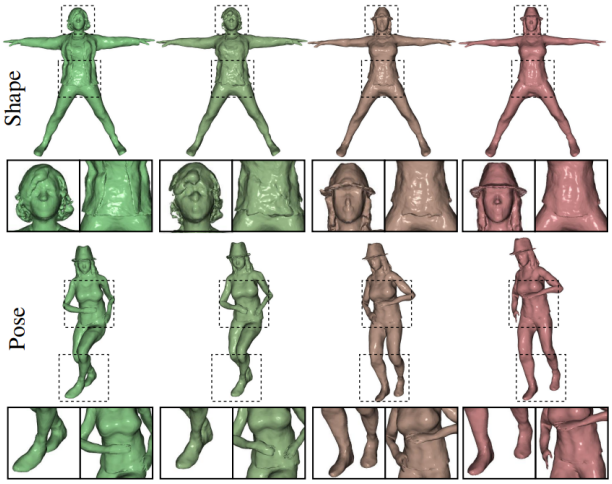

Figure 3: Interpolation of shape (top) and pose (bottom). From Palafox, Božič, et al. (2021).

As an important limitation, NPMs cannot be controlled directly or in a semantically meaningful way. Traditional parametric models assign a semantic meaning to all of their parameters. Considering the SMPL model, for instance, each scalar parametric component is a linear coefficient of a basis of distinct shape characteristics. Extracted by PCA, these can be interpreted with regard to properties such as body height, body mass, and combination thereof, affected by real-world correlations. Using NPMs on the other hand, a targeted manipulation of the shape and poses of identity i could only be performed indirectly by generating a sequence of meshes where the desired shape and poses are observable, fitting a fully trained model to it, and transferring the corresponding codes to identity i.

This lack of interpretability also motivates why NPMs are not considered a suitable model for generative tasks. In general, any code from the learned space of shape can be translated into the mesh of a feasible shape in its canonical pose, and any code from the pose space can be interpreted as a valid deformation field. Since the shape is not constructed explicitly, from a user's perspective, it is not possible to manually make assumptions about the output of the MLPs, neither to generate parametric codes without performing a costly test-time inference. Even though it is feasible to synthesize novel shapes or poses from NPMs, their lack of control is a great disadvantage compared to established alternatives.

Although reconstructing hair and clothing wrinkles is achieved only to some extend, the experiments of Sec. 4 demonstrate that NPMs are particularly suitable for fitting a model to unseen observations of a deforming shape. However, there are some limitations with regard to the expressiveness. According to Palafox, Božič, et al. (2021), inaccuracies in the implicit representations can lead to drastic effects in low-volume areas, e.g., ties. Furthermore, the behavior of loose clothing is hard to predict and remains challenging to capture. In addition, one might argue that certain areas have a higher significance than others (e.g., faces), a perspective that is not considered in the construction of NPMs. Additionally, NPMs employ basic MLP architectures. More sophisticated architectures might increase their accuracy.

As presented throughout this report, the main strengths of the approach of NPMs revolve around its flexibility due to the learning-based construction process. NPMs can model

non-rigid shapes of any domain without the need for manual annotations or domain-specific knowledge. Consequently, they do not require various model extensions to account for further details (e.g., clothes, hair). They reconstruct objects of arbitrary topologies combining a continuous representation and detail.

However, the method makes several assumptions about the training data. Specifically, the train set needs to include each identity as a template shape in a common pose, i.e., the canonical pose. Each identity needs to be present with a sufficient amount of pose variations to learn the pose space that comes with a large number of degrees of freedom. Moreover, there needs to exist dense surface correspondences between these and the canonical shape, i.e., exact registrations of the canonically posed mesh to the posed ones. This is commonly achieved by registering a template-based model to real-world 3D scans. Both the use of such a model and the specification of a canonical pose can be regarded as indirect dependencies on manual annotations.

During review of the literature, it has become apparent that identifying models that are learned, not domain-specific, can handle non-rigid shapes in motion, and make similar assumptions about the training data as NPMs, is difficult. Within the considerations in this report, it remained unclear how to evaluate the performance of NPMs with different kinds of models considering these differences. Moreover, a comparison of their fitting accuracy is often obstructed because they employ different evaluation metrics and input types for fitting (e.g., partial or complete 3D information, sequences, or single frames).

As part of this report, a minimalistic experimental setup was created to test the fitting capabilities of NPMs. Using a pretrained model of human bodies, the input data was adapted to contain only a single depth map of a single sequence of CAPE. As an observation, the accuracy might be slightly impaired from restricting the input to a single viewpoint in time (IoU of 0.81). In addition, it has become apparent that the computational time required by the fitting process is several orders of magnitudes greater than for basic models such as SMPL. Because results on human bodies are already available in the original work, no further investigations have been performed. The configuration adaptations and small own scripts for visualization can be examined at `https://github.com/bertan-karacora/npms`.

While NPMs outperformed comparable state-of-the-art methods for model fitting to partial 4D data at the time, in more recent work, several extensions have been proposed which suggest that separating objects into parts or limbs before modeling these with NPMs leads to a higher fitting accuracy, e.g., SPAMs (Palafox, Sarafianos, et al., 2022). In a sense, one might argue that this puts class-specific constraints back into the model. This raises questions about the degree to which these models can be constrained in a useful way that not only improves the accuracy but also human intuition, an aspect that warrants further exploration.

6 Conclusion

In this report, the approach of Neural Parametric Models is presented. Proposed by Palafox, Božič, et al. (2021), NPMs offer an innovative perspective on the construction of dynamic models for 3D deformable shapes such as human bodies or hands. Opposed to traditional parametric models that rely on an inflexible parametrization, NPMs can be learned from data alone which makes this method applicable to any class of non-rigid shapes. To achieve an accurate reconstruction and tracking of 4D dynamics, NPMs combine two key ideas: First, disentangling shape and pose effectively into separate parameter spaces, and second, leveraging continuous implicit functions to represent both the shape's surface and its pose-dependent deformations. Following the auto-decoder paradigm, this method creates latent spaces of shape and pose which show well-regularized properties enabling accurate fitting via inference-time optimization as well as smooth latent space interpolation. Importantly, NPMs lack interpretability of the individual scalar parametric components entirely which impedes the control, manipulation, and analysis of registered shapes.

References

Alldieck, T., M. Magnor, B. L. Bhatnagar, C. Theobalt, and G. Pons-Moll (2019). "Learning to Reconstruct People in Clothing from a Single RGB Camera". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Anguelov, D., P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis (2005). "SCAPE: Shape Completion and Animation of People". In: *ACM SIGGRAPH 2005 Papers*. SIGGRAPH '05, pp. 408–416.

Bhatnagar, B. L., C. Sminchisescu, C. Theobalt, and G. Pons-Moll (2020). "Combining Implicit Function Learning and Parametric Models for 3D Human Reconstruction". In: *European Conference on Computer Vision (ECCV)*.

Cao, Z., G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh (2019). "OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Chen, X., T. Jiang, J. Song, J. Yang, M. J. Black, A. Geiger, and O. Hilliges (2022). "gDNA: Towards Generative Detailed Neural Avatars". In: *arXiv preprint arXiv:2201.04123*.

Giebenhain, S., T. Kirschstein, M. Georgopoulos, M. Rünz, L. Agapito, and M. Nießner (2023). "Learning Neu-

ral Parametric Head Models". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jiang, B., Y. Zhang, X. Wei, X. Xue, and Y. Fu (2022). "H4D: Human 4D Modeling by Learning Neural Compositional Representation". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Li, Y., H. Takehara, T. Taketomi, B. Zheng, and M. Nießner (2021). "4dcomplete: Non-rigid motion estimation beyond the observable surface". In: *IEEE International Conference on Computer Vision (ICCV)*.

Loper, M., N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black (2015). "SMPL: A Skinned Multi-Person Linear Model". In: *ACM Trans. Graphics (SIGGRAPH Asia)* 34.6, 248:1–248:16.

Lorensen, W. E. and H. E. Cline (1987). "Marching Cubes: A High Resolution 3D Surface Construction Algorithm". In: *14th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '87, pp. 163–169.

Ma, Q., J. Yang, A. Ranjan, S. Pujades, G. Pons-Moll, S. Tang, and M. J. Black (2020). "Learning to Dress 3D People in Generative Clothing". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Mahmood, N., N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black (2019). "AMASS: Archive of Motion Capture as Surface Shapes". In: *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5442–5451.

Mohamed, M. and L. Agapito (2022). "GNPM: Geometric-Aware Neural Parametric Models". In: *arXiv preprint arXiv:2209.10621*.

Mu, J., W. Qiu, A. Kortylewski, A. Yuille, N. Vasconcelos, and X. Wang (2021). "A-SDF: Learning Disentangled Signed Distance Functions for Articulated Shape Representation". In: *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12981–12991.

Niemeyer, M., L. Mescheder, M. Oechsle, and A. Geiger (2019). "Occupancy Flow: 4D Reconstruction by Learning Particle Dynamics". In: *IEEE/CVF International Conference on Computer Vision (ICCV)*.

Osman, A. A. A., T. Bolkart, and M. J. Black (2020). "STAR: A Sparse Trained Articulated Human Body Regressor". In: *European Conference on Computer Vision (ECCV)*, pp. 598–613.

Palafox, P., A. Božič, J. Thies, M. Nießner, and A. Dai (2021). "NPMs: Neural Parametric Models for 3D Deformable Shapes". In: *IEEE/CVF International Conference on Computer Vision (ICCV)*.

Palafox, P., N. Sarafianos, T. Tung, and A. Dai (2022). "SPAMs: Structured Implicit Parametric Models". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Park, J. J., P. Florence, J. Straub, R. Newcombe, and S. Lovegrove (2019). "DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Pavlakos, G., V. Choutas, N. Ghorbani, T. Bolkart, A. A. A. Osman, D. Tzionas, and M. J. Black (2019). "Expressive Body Capture: 3D Hands, Face, and Body from a Single Image". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10975–10985.

Romero, J., D. Tzionas, and M. J. Black (2017). "Embodied Hands: Modeling and Capturing Hands and Bodies Together". In: *ACM Transactions on Graphics, (SIGGRAPH Asia)*. 245:1–245:17 36.6.

Tian, Y., H. Zhang, Y. Liu, and L. Wang (2022). "Recovering 3D Human Mesh from Monocular Images: A Survey". In: *arXiv preprint arXiv:2203.01923*.

Tretschk, E., N. Kairanda, M. B R, R. Dabral, A. Kortylewski, B. Egger, M. Habermann, P. Fua, C. Theobalt, and V. Golyanik (2023). "State of the Art in Dense Monocular Non-Rigid 3D Reconstruction". In: *Computer Graphics Forum (Eurographics State of the Art Reports)*.

Wei, F., R. Chabra, L. Ma, C. Lassner, M. Zollhoefer, S. Rusinkiewicz, C. Sweeney, R. Newcombe, and M. Slavcheva (2022). "Self-supervised Neural Articulated Shape and Appearance Models". In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Xie, Y., T. Takikawa, S. Saito, O. Litany, S. Yan, N. Khan, F. Tombari, J. Tompkin, V. Sitzmann, and S. Sridhar (2022). "Neural Fields in Visual Computing and Beyond". In: *Computer Graphics Forum*.