NIPIG: Neural Implicit Avatar Conditioned on Human Pose, Identity and Gender

Guillaume Loranchet
InterDigital, Inria, University Rennes, IRISA
Rennes, France
guillaume.loranchet@InterDigital.com

Pierre Hellier
University Rennes, Inria, CNRS, IRISA
Rennes, France
pierre.hellier@univ-rennes1.fr

Adnane Boukhayma
Inria, University Rennes, CNRS, IRISA
Rennes, France
adnane.boukhayma@inria.fr

João Regateiro
InterDigital
Rennes, France
Joao.Regateiro@InterDigital.com

Franck Multon
Inria, University Rennes, CNRS, IRISA
Rennes, France
Franck.Multon@irisa.fr

ABSTRACT

The creation of realistic avatars in motion is a hot-topic in academia and the creative industries. Recent advances in deep learning and implicit representations have opened new avenues of research, especially to enhance the details of the avatars using implicit methods based on neural networks. State-of-the-art implicit Fast-SNARF [Chen et al. 2023] methods encodes various poses of a given identity, but are specialized for that single identity. This paper proposes NIPIG, a method that extends Fast-SNARF model to handle multiple and novel identities. Our main contribution is to condition the model on identity and pose features, such as an identity code, a gender indicator, and a weight estimate. Extensive experiments led us to a compact model capable of interpolating and extrapolating between training identities. We test several conditioning techniques and network's sizes to find the best trade-off between parameter count and result quality. We also propose an efficient fine-tuning approach to handle new out-of-distribution identities, while avoiding decreasing the reconstruction performance for in-distribution identities.

CCS CONCEPTS

• **Computing methodologies** → **Shape modeling**; **Animation**.

KEYWORDS

Computer graphics, Character Animation, Implicit methods, Generative models

ACM Reference Format:
Guillaume Loranchet, Pierre Hellier, Adnane Boukhayma, João Regateiro, and Franck Multon. 2025. NIPIG: Neural Implicit Avatar Conditioned on Human Pose, Identity and Gender. In *European Conference on Visual Media Production (CVMP '25), December 3–4, 2025, London, United Kingdom.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3756863.3769702

CVMP '25, December 3–4, 2025, London, United Kingdom

ACM ISBN 979-8-4007-2117-5/2025/12…$15.00
https://doi.org/10.1145/3756863.3769702

1 INTRODUCTION

The digitization of humans in motion has become a critical topic in the digital creative industry for applications in gaming, VFX, and try-on marketing. Digital representations of humans (avatars) are expected to be high-quality, animatable, and interoperable across platforms. In VR social networks and telepresence, avatars should support real-time streaming for communication systems with a trade-off between high-quality representation and memory usage. Despite recent advancements, accurately modeling, reproducing, and controlling the complex deformations and anatomical shapes involved in human motion, in a compact manner, remains challenging. In massively immersive or gaming virtual worlds inhabited by avatars, transferring the accurate animated body shape still leads to massive data traffic, which is incompatible with sustainable growth.

Explicit compact representations of humans and poses exist, but they are generally limited to naked human bodies and approximated human body templates. Explicit models are based on a fixed sampling of surface vertices, with high memory usage to handle fine details. A parallel branch of research explored implicit representations of animated human bodies. Hence, instead of using traditional explicit representation based on mesh deformation, these approaches explored the use of a single efficient and lightweight function to model human body shape. It enables to encode more or less fine details with the same implicit function.

Implicit models have proven to be an appealing alternative approach to represent static scenes with arbitrary topologies as a continuous and resolution-agnostic function. More recently, neural implicit models [Mescheder et al. 2018; Park et al. 2019] improved the representation of complex shapes by leveraging neural networks' capacity to fit training data. Similarly, implicit neural models [Chen et al. 2023, 2021] have been used to learn an implicit representation of an animated avatar with minimal clothing, often helped by traditional explicit animation techniques such as the Linear Blend Skinning (LBS). These implicit representations enable reliable and efficient animation of realistic digital humans, even in poses far from the learned pose distribution. The implicit model can be trained using high-quality public datasets. However, these implicit models rely on complex and huge neural network architectures with numerous parameters, which makes them prohibitive for streaming applications in terms of size and speed. Moreover,

these implicit models are trained on a single shape identity (or morphology), which generally requires retraining and disseminating a complete model for each new avatar appearance.

In this paper, we propose NIPIG, a method that generalizes implicit models to unseen poses and identities. This leads to a unique generative model, controlled by a lightweight identity code, that compactly represents many identities. We have performed extensive evaluations to design the best compromise between 3D human shape reconstruction accuracy and reduced size of the model, in term of parameter count. When introducing a compact identity latent code at the optimal position in the architecture, the model's size remains similar to the one of Fast-SNARF [Chen et al. 2023], with similar accuracy. However, the model can generalize to a large number of possible identities. For characters out of the training set distribution, a fine-tuning process enables accuracy comparable to single identity models with far less training. In this paper, we define out-of-distribution avatars as characters that cannot be approximated by interpolating subjects in the training set. Finally, we compared our results with COAP: compositional articulated occupancy of people [Mihajlovic et al. 2022]. COAP [Mihajlovic et al. 2022] is a multi-part implicit avatar representation able to quickly handle self-intersection. This model is the state-of-the-art in representing multi-identity implicit body surface.

To summarize, NIPIG is a model that is lighter than state-of-the-art explicit parametric model (SMPL [Loper et al. 2015]), whilst achieving better quality than state-of-the-art implicit methods. Although slower than SMPL [Loper et al. 2015], the proposed model is 38 times smaller in memory (76 MB for SMPL male+female models versus 2 MB for NIPIG).

This model is also not limited to SMPL-compliant shapes. For example, it can learn its own identity latent space and, when conditioned on a gender code, it can linearly interpolate between male and female.

The main contributions of this paper are:

- A lightweight implicit model for a neural avatar, conditioned on identity and gender latent codes, able to generate new poses and identities on the fly,
- An efficient fine-tuning method to add new identities to the generative models,
- A thorough analysis of the model architecture and identity conditioning to determine the optimal trade-off between reconstruction quality and model size.

2 RELATED WORK

In this section, we present the closest related work to this paper. We start with a review of explicit models before focusing on implicit neural avatars.

2.1 Explicit avatars

Explicit deformations of a mesh can be performed by combining blend skinning techniques and parametric models [Hirshberg et al. 2012; Loper et al. 2015], which enables the generation of new posed identities. Widely used by the community as an explicit parametric representation, the SMPL model [Loper et al. 2015] disentangles a given frame into two parts: its pose and its shape (or identity/morphology). The shape parameter is a relatively small vector of PCA coefficients learned on a dataset of hundreds of subjects in a canonical pose. By the property of the PCA, these components can then be linearly interpolated to generate an infinite number of new identities. The pose parameters are composed of all rotation matrices defining a given pose. These parameters are used as coefficients for the blend shapes. Thus, each pose in SMPL is a blending of many real scans. Once the template mesh is locally deformed in canonical space based on the identity and pose parameters, it is finally reposed with Linear Blend Skinning (LBS) or Dual Quaternion Skinning (DQS) [Kavan et al. 2007]. However, the SMPL model, by design, is limited to a fixed topology and requires transferring a large set of weights to store all the different deformed shapes.

More recently, 3D Gaussian Splats (3DGS) [Kerbl et al. 2023] are proposed as a new representation of avatars, including clothed people using parameterized models [Li et al. 2024; Moon et al. 2024]. Previous works [Li et al. 2024] represent the human body using a model parameterized by a LBS function, which modifies the 3DGS spatial positions with a Multi-Layer Perceptron (MLP). This method also includes conditioning between adjacent 3DGS in both deformed and canonical pose spaces to maintain physical rigidity while permitting some non-rigid deformation. ExAvatar [Moon et al. 2024] aims to create expressive full-body avatars by integrating the SMPL-X [Pavlakos et al. 2019] body model with 3DGS, using monocular videos. It tackles scenarios with limited data, such as facial expressions without ground truth, by aligning the SMPL-X model to images using a regressor and a 2D pose estimator. For facial features, DECA [Feng et al. 2021] and the FLAME [Li et al. 2017] model are used to extract shape identity and expressions. However, the model is intricate and multi-modal. In this work we focus on surface-only implicit geometry, not appearance. Furthermore, 3DGS models often require many parameters thus taking up much more space.

2.2 Implicit avatars

A seminal implicit-avatar model is NASA [Deng et al. 2020]. It predicts the occupancy [Mescheder et al. 2018] of each body part given bone transformations. However, modeling body parts separately can lead to visible artifacts at the intersections [Deng et al. 2020] [Mihajlovic et al. 2022], especially for unseen poses. Furthermore, it requires the ground truth skinning weight at both learning and inference time, which limits generalization [Chen et al. 2025].

COAP [Mihajlovic et al. 2022] improved on NASA [Deng et al. 2020] by making each body part deformation conditioned on its neighboring parts. This mitigated the artifacts between body parts without completely fixing the issue. Regarding full body prediction, Neural GIF [Tiwari et al. 2021] and LEAP [Mihajlovic et al. 2021] learned a separate skinning network to map points from pose to canonical space. They could generalize to unseen poses but fail under extreme conditions due to their backward skinning. Backward skinning is the process of learning an inverse skinning field to map points from pose space to canonical space.

Pose-conditioned networks can predict inverse skinning weights but are very sensitive to small changes in the pose space. This sensitivity leads to challenging training and often requires ground truth skinning weights for supervision [Saito et al. 2021]. Furthermore, this additional pose-conditioning hinders the generalization

capabilities of the network as it struggles to predict the skinning weights of unseen poses.

Finally, as stated in Fast-SNARF [Chen et al. 2023], using a pose-dependent skinning field reduces the possibilities of incorporating a voxel grid-like acceleration data structure, as they require a smooth and stable space for interpolation. Due to these limitations concerning backward skinning, SNARF [Chen et al. 2021] (and later Fast-SNARF) chose only to implement forward skinning, defined in canonical space. Thus, they were not dependent on the pose. To do so without relying on ground truth skinning weights, they first find several initial canonical roots (one for each of the selected bones) for each pose point by deforming it rigidly. Then, they iteratively optimize the positions of the points in canonical space such that, given a set of skinning weights predicted by an MLP, the newly found pose point is the same as the initial query point.

Like most other implicit methods, SNARF conditions the Occupancy network on the pose parameter to help the model learn non-linear deformations associated with certain poses. To optimize the querying speed of the skinning weight field, Fast-SNARF replaces the initial neural network with a voxel grid. The skinning weights are only defined for the grid points and each query point is then trilinearly interpolated to obtain its final weights. A shallow MLP parameterizes the voxel grid to ensure smoothness. However, unlike SNARF, it only has to predict values for the grid points, thus greatly increasing the computational speed.

As reported in a recent survey [Gu et al. 2025], Fast-SNARF presented a state-of-the-art fast and pose generalized method for implicit avatar representation. It has been recently used to create animatable human avatars from motion sequences [Huang et al. 2024]. Other methods used sequences of RGB-D images [Dong et al. 2022][Wang et al. 2021] or a single RGB-D image [Pesavento et al. 2024]. More recent works have proposed alternative approaches to handle complex clothing [Zhu et al. 2024]. However, most of these methods involve training a subject-specific model, requiring to train a new model for each new identity.

Methods like gDNA [Chen et al. 2022] and SMPLicit [Corona et al. 2021] tackled the issue of shape generalization but focused on clothed subjects and required numerous neural networks, leading to large computation and memory complexities. Furthermore, gDNA [Chen et al. 2022] used one shape per identity and cannot associate different shapes to a single identity, thus lacking precise identity control. Most state of the art implicit methods like COAP and Neural-ABC [Chen et al. 2025] heavily rely on a parametric model such as SMPL, even even at inference. This dependency limits the model to generate the same shape as SMPL, as well as increasing the total number of parameters required. In contrast, our method does not depend on SMPL model at inference time. To our knowledge, we are the first to propose a lightweight implicit model with forward skinning that generalizes to both pose and identity.

3 FAST-SNARF OVERVIEW

In this section, we recall the method presented in Fast-SNARF [Chen et al. 2023] as our approach is heavily based on it. The key idea of Fast-SNARF [Chen et al. 2023] is to use only forward skinning to generalize to unseen poses. The occupancy is predicted in the canonical space which allows to easily apply any rigid deformation thanks to the Linear Blend Skinning (LBS). To improve the generalization, Fast-SNARF assumed that the skinning weights are unknown during training. Additionally, most datasets contain poses, but without the corresponding canonical pose. For these two reasons, [Chen et al. 2023] chose to query points in pose space and find their canonical correspondences iteratively. To do so, they first rigidly deformed the pose points to the canonical space. They then trained a network S that predicts their skinning weights. Given those weights, the position of the canonical points is iteratively updated such that they are mapped to the desired point in pose space. The canonical points are attributed an occupancy probability with a second network O, which can be compared to the ground truth occupancy of the query point computed in advance. Pre-computing a dataset of points with their corresponding occupancy saves a lot of time during training. At inference time, the occupancy is used by the marching cubes algorithm to find the surface of the body. We now first present the fast forward-skinning model that efficiently computes LBS weights (see section 3.1). In a second part, we present the occupancy network (see section 3.2).

3.1 Fast forward skinning

As a reminder, Fast-SNARF [Chen et al. 2023] improves SNARF [Chen et al. 2021] by representing the linear blend skinning field as a voxel grid. Indeed, the network S aims at predicting the skinning weights for any point in $3D$. However it would require to query the network for each point, and for each frame, for training and inference. Instead, Fast-SNARF [Chen et al. 2023] proposed to speed up the process by only predicting the weights of a fixed number of points located on a voxel grid. The skinning weight of any canonical point $x_c \in \mathbb{R}^3$ can then be trilinearly interpolated from the grid (see Figure 2). As the trilinear interpolation and the transformation T are both linear operations, they first precompute the transformation matrix for each grid point before interpolation. The final interpolated transformation is used to compute the new position x_p in pose space of any canonical point x_c.

For each grid point $x_g \in \mathbb{R}^3$, the network S predicts its skinning weight $w_g \in \mathbb{R}^{n_b}$. Given the set of bone transformations $B = \{B_1, ..., B_{n_b} | B_i \in SE(3)\}$ with n_b the number of bones, we can define the following transformation:

$$\mathcal{T}(x_g, B) = \sum_{i=0}^{n_b} S(x_g)_i \cdot B_i \tag{1}$$

With trilinear interpolation, each canonical point x_c thus has the transformation: $\mathcal{T}(x_c) = \text{Trilerp}(\mathcal{T}(\mathcal{N}_1), ..., \mathcal{T}(\mathcal{N}_8))$ where $\mathcal{N}_i$ is the i-th neighbor of x_c in the voxel grid. This yields the posed position x_p of x_c as: $x_p = \mathcal{T}(x_c) \cdot x_c$

The previous transformation maps points from canonical to pose spaces but also requires the inverse mapping, as points are queried in the deformed space. However, this non-bijective function is not invertible: one posed point can have several canonical points in case of self contacts. Fast-SNARF [Chen et al. 2023] defines the following function $f(x_c) = T(x_c) \cdot x_c - x$ that should be equal to 0 for the correct canonical points associated to the query point x:

It enables to iteratively find the multiple solutions to $f(x_c) = 0$ with a quasi-Newton algorithm, with various starting points for

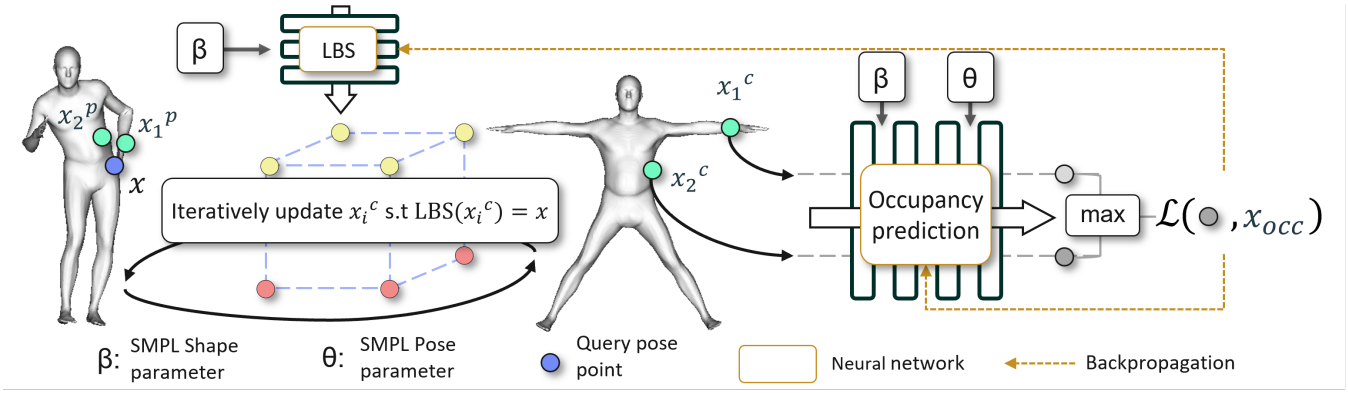


Figure 1: General overview: For any query point x in pose-space, the model first finds iteratively the corresponding roots $\{x_1, ...x_n\}$ in the canonical space thanks to the LBS network, conditioned on the identity vector β. Then, for each canonical root, a second network predicts its occupancy conditioned on the identity (β) and pose (θ). The result is finally compared with the ground truth occupancy of x with a Binary Cross Entropy loss $\mathcal{L}$ to update both networks.

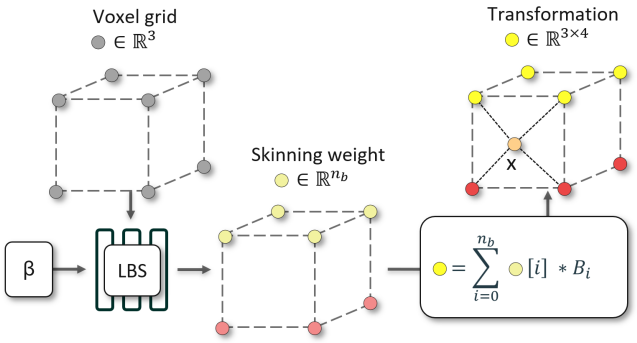


Figure 2: Overview of the LBS network. A neural network, conditioned on the identity β, first predicts skinning weights of dimension R^{n_b} for every points in a voxel grid. For those same points, a transformation is computed . Finally, the transformation of any point in 3D can be evaluated with trilinear interpolation.

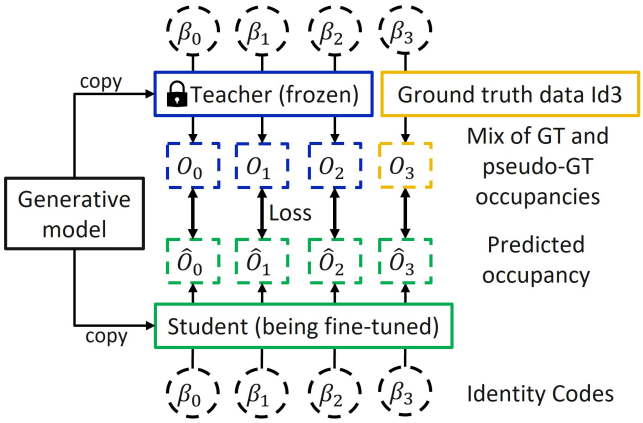


Figure 3: Overview of the fine-tuning. The original generative model is split into two: the teacher model predicts pseudo ground truth occupancies (O_{0-2}) while the student model is fine-tuned on those predictions combined with the original ground truth data (O_3). Both of them are conditioned on the same identity codes β_i

the initialization. Fast-SNARF [Chen et al. 2023] selects one point per body part and rigidly deforms it to the canonical space for initialization.

3.2 Occupancy prediction

Once the canonical correspondences $\{x_1, ...x_n\}$ are found, they each get assigned an occupancy probability $o_i \in \mathbb{R}$ by the neural network $O(x_i, \theta) = o_i$.

Here, $\theta \in \mathcal{Y} \subset \mathbb{R}^{3\times n_b}$ is the SMPL [Loper et al. 2015] pose parameter, a $3 \times n_b$ set of bone rotation vectors. In Fast-SNARF [Chen et al. 2023], the occupancy network is only conditioned on θ. The final result is given as the maximum of the previous occupancies. A binary cross entropy loss is used with the ground truth occupancy. Finally, all the network weights are updated jointly.

4 MULTI IDENTITIES

In this section, we first present our overall approach, before focusing on the architecture of the networks, the different tested conditioning methods, and the training procedure used. Finally, the last subsection is dedicated to the fine-tuning. In Fast-SNARF [Chen et al. 2023], neither network S nor O are conditioned on identity. It thus requires to train new networks S and O for each new character. We hypothesize that by conditioning both networks on the identity at the optimal location in the network, it is possible to model several persons, and even generate new ones without increasing the size of the networks, nor losing quality. Figure 1 presents our overall pipeline.

The networks can be conditioned with different identity representations. We experimented with either using the SMPL [Loper

et al. 2015] identity vector β, or letting the model learn its own latent code. In the second case, similar to β, each identity latent code is of size 10.

We found almost identical results with the two approaches. It is important to note that, unlike almost all competitors, NIPIG can thus work independently of SMPL. However, in most experiments, we chose to use β to simplify the comparison with the ground truth SMPL model. The network S, conditioned on β, gives $S(x, \beta) = w$. The transformation matrices and therefore the reposing of canonical points now depend on the identity.

In a similar manner, the occupancy network becomes $O(x, \theta, \beta) = o$. To represent both male and female characters, we added a single value to the SMPL [Loper et al. 2015] identity vector: 0 for males and 1 for females. We use a binary gender code as it is the default separation in the dataset with use. Although the model is trained with a binary code, at inference, it is able to smoothly interpolate given a continuous gender code.

4.1 Architecture

The skinning network is composed of 4 fully connected layers of size 128 with softplus activation for the first 3 layers. The output skinning weights are given in a vector of size $n_b = 24$. The architecture of the occupancy network is similar, but with 8 layers instead of 4. All layers are of size 256, except the one conditioned on the pose, which has an input size of 264. Indeed, the pose parameter is embedded with a single layer of output size 8, and added at the 6th layer. The input, the coordinates of the $3D$ point to query, is fed again in the middle of the occupancy network, as a skip connection, to stabilize training, similarly to Fast-SNARF. We also show the results obtained with a much smaller model, composed of layers of size 128 for the occupancy network (instead of 256) and 64 for the skinning network (instead of 128). This lighter model has almost 4 times less parameters, from 522 k for the original model to 134 k.

4.2 Conditioning

We experimented with different identity-conditioning methods: concatenation, FiLM (Feature-wise Linear Modulation) [Perez et al. 2017], and hypernetwork [Ha et al. 2016]. We aim at finding the most expressive conditioning method for a minimal number of parameters.

4.2.1 Concatenation. For any layer index l, let $x_{l+1} \in \mathbb{R}^{n_{l+1}}$ the output of this layer be defined as $x_{l+1} = f_l(x_l)$, with $f_l(x_l) = \text{softplus}(W_l \cdot x_l + b_l)$. Then let i be the index of the layer to condition on the identity. We obtain:

$$x_{i+1} = f_i\left([x_i; f_c(c)]\right)$$

with $c \in \mathbb{R}^{10}$ the identity code and $f_c(c) = \tanh(W_c \cdot x_c + b_c)$ a single layer to encode the identity. Here, $[x_i; f_c(c)] \in \mathbb{R}^{n_i+m}$ with m the embedded size of the identity code (in our experiment, we use $m = 8$). This method is straight forward and adds a limited number of parameters

4.2.2 FiLM. In this case, instead of concatenating, we apply a modulation on the entire feature vector x_i, giving:

$$x_{i+1} = f_i\left(\gamma(c) \cdot x_i + \delta(c)\right)$$

with $\gamma(c)$ and $\delta(c)$ in $\mathbb{R}^{n_i}$, single linear layers with no activation. Thus $\gamma(c) \cdot x_i + \delta(c)$ with the element-wise multiplication and addition is also in $\mathbb{R}^{n_i}$. This method impacts the feature vector more globally, with a small added cost.

4.2.3 Hypernetwork. Finally, a hypernetwork predicts directly the weights and biases of layer i, conditioned on the identity code. We have:

$$x_{i+1} = \text{softplus}(W(c) \cdot x_l + b(c))$$

with $W(c)$ and $b(c)$ mapping c to respectively W_i and b_i. Thus $W(c)$ is a matrix of size $10 \times (n_{i-1} \times n_i)$ and $b(c)$ is a vector of size $10 \times n_i$. The hypernetwork is effective as it generates the full layer i based on the identity. However, even by applying it on only one layer, the number of parameters is more than doubles.

4.3 Training

Both S and O networks are trained simultaneously. Similarly to Fast-SNARF [Chen et al. 2023], for each frame, we sample $100k$ points uniformly in space ($\mathcal{D}_u$), and another $100k$ points close to the surface of the mesh ($\mathcal{D}_s$). Their occupancy is pre-calculated and stored beforehand to reduce the computational cost during training. It is sufficient to compare the predicted and ground truth occupancy. Generating the entire mesh would not add more information, while taking a lot of time. During training, at each iteration, 250 points are drawn from $\mathcal{D}_u$, while 2000 come from $\mathcal{D}_s$. As we are training on more subjects than Fast-SNARF, we increase the number of iterations from 45k to 150k.

Similarly to Fast-SNARF, the main loss is a Binary Cross Entropy (BCE) between the predicted and known occupancy. Fast-SNARF also added two secondary losses to guide the networks at the beginning of the training: the first one is a Mean Squared Error (MSE) loss to ensure that the predicted skinning weight at joints position is one only for that joint. The second one is a BCE to force the occupancy prediction of points on the bones to be 1. The final total loss is:

$$\mathcal{L} = \lambda_b \mathcal{L}_{\text{bone}} + \lambda_j \mathcal{L}_{\text{joint}} + \lambda_{BCE} \mathcal{L}_{BCE}, \tag{2}$$

$$\mathcal{L}_{\text{bone}} = BCE(O(x_{\text{bone}}, \beta, \theta), 1), \tag{3}$$

$$\mathcal{L}_{\text{joint}} = \sum_{j=0}^{|B|-1} \left(S(x_j, \beta, \theta) - e_j\right)^2 \tag{4}$$

Where $|B|$ is the number of bones, x_j is the coordinate of the joint j, and e_j is the one hot vector consisting of only zeros and a one at index j. λ_b is set to 1 and λ_j to 10.

Some other aspects have been improved compared to Fast-SNARF. We use the AdamW optimizer with the CosineAnnealingLR scheduler from PyTorch, making the learning rate oscillate between $1e-3$ and $2e-5$. We also add a Stochastic Weight Average (SWA) module at the end of the training. This module averages the learnable weights over several iterations to obtain a more stable model. Finally, we changed the points sampling strategy in the preprocessing step. Fast-SNARF sampled points uniformly on the mesh. Instead, we sample the same number of points for each face. Thus, more points are sampled on higher density region such as the face.

4.4 Fine-Tuning

Through identity conditioning, NIPIG is able to generate a whole set of people, by navigating in the identity latent space. However, given the huge diversity of real-life identities, it is impossible to guarantee that each one of them can be represented by the model. It is thus important to be able to add a new identity to the network without corrupting its current knowledge, which is not trivial. In other words, our goal is to update the model with a new identity to improve its generalization. Naively, a very inefficient method could be to train again the model from scratch with the added identity. A better approach would be to only fine-tune the model on the new identity. However, to avoid the model forgetting the previous identities, concepts often called catastrophic forgetting, one needs to add them to the fine-tuning. This method requires storing the training data of all identities, which in practice, is not feasible. Instead, we propose to use the knowledge of the pretrained generative model as supervision during the fine-tuning. In this case, we only require two copies of the model's weights, much smaller than the training set. An overview of the method is described in Figure 3. More precisely, before the fine-tuning, the model is duplicated into two parts: the teacher $\mathcal{M}_t$ and the student $\mathcal{M}_s$. Both parts predict the occupancy of points in pose space x_p. The teacher is frozen and used as supervision. To fine-tune the student model, similarly to the original training, we select a set of points extracted from the fine-tuned character's dataset. At each step, we create a batch of $\beta \sim B$ randomly sampled from B, the set of all βs. B is composed of both the β of the fine-tuned character (β_{New} in eq 5) and of the βs of the previous characters (β_{Prev} in eq 5). When the model selects the new character's β, the student tries to predict the ground truth occupancy o_p. When conditioned on previous identities, the student must predict results similar to the teacher model. Thus, we force the student model to remember previous predictions, without needing the entire training set.

The final loss is given as:

$$\mathcal{L} = \sum_{\beta \in B_{\text{New}}} BCE(\mathcal{M}_s(x_p, \beta, \theta), o_p) + \sum_{\beta \in B_{\text{Prev}}} BCE(\mathcal{M}_s(x_p, \beta, \theta), \mathcal{M}_t(x_p, \beta, \theta)) \tag{5}$$

5 EXPERIMENTAL SET-UP

In the following experiments, we have tested different architectures and conditioning options to obtain the best compromise between the ability to generalize to new identities and the minimal network size. In this section, we present the datasets and metrics used to perform these experiments.

5.1 Datasets

During our experiments, we have used the DFaust [Bogo et al. 2017] dataset as we focus on minimally clothed avatars. DFaust consits of 10 subjects (5 females and 5 males) each performing around 10 actions. For fairness, we have selected the same two actions for all characters for validation and testing: "running on spot" and "punching" as they showcase the most diverse movements, for the upper and lower bodies. For training, for each character, we have selected 9 actions. We selected similar actions where possible.

We computed our metrics on three different dataset splits. The first one, denoted "training data", is the whole training dataset. The second one, denoted "unseen poses", is composed of 2 sequences left out of the training set: "running on spot" and "punching". The third one, denoted "unseen identities and poses", contains the same two actions mentioned above, but recomputed for a new set of identities by modifying the SMPL [Loper et al. 2015] parameters. To create new identities, we have made pairs of characters and averaged their corresponding β vectors. For example, using 5 characters, we have generated 10 new identities thanks to interpolation. The βs and the pose parameters of the 2 left-out actions were used to create a new dataset of unseen identities and poses.

We are also interested in training the model on more identities to study its representation capabilities. However, finding a diverse dataset can be tedious. We thus decided to create a synthetic dataset thanks to SMPL's β parameter. The previously mentioned new identities are only used to evaluate the ability of the model to interpolate between known identities. They are quite similar to the original DFaust characters, and therefore not relevant for our extended dataset. Instead of simply interpolating the βs from the DFaust character, we sampled new ones from a multivariate normal distribution. The mean is zero for all components except for the first one. We have found empirically that we could obtain more realistic shapes with the first component set to -0.5 for males and 0.7 for females. Finally, the variance is set to 1.75. We sampled 28 new identities, with heights ranging from 1.42m to 1.96m.

We have also computed results for unseen movements, such as the "AIST" [Li et al. 2021] demo dance sequence (see Figure 4 for a single frame, and a video is given in the supplementary materials). Finally, we added sequences from the challenging MPI PosePrior dataset [Akhter and Black 2015], which is far from the training set distribution (see the supplementary materials).

5.2 Metrics

Two metrics have been used to evaluate our results quantitatively: intersection over union, and surface distance.

Intersection over Union. Similarly to Fast-SNARF [Chen et al. 2023], we used the Intersection over Union (IoU) of correctly predicted occupancy points, for both points sampled close to the surface (IoU surf) and for points sampled uniformly (IoU bbox).

Surface distance. We also computed a Chamfer distance between the simulated and the ground truth meshes. We uniformly sampled 20 k points on the surface of the simulated mesh and computed the distance of each point to the closest face in the ground truth mesh. This way, the comparison was not constrained by topological differences between the two meshes.

6 RESULTS

This section presents both qualitative and quantitative evaluations of the proposed method, alongside comparisons with the state-of-the-art techniques, Fast-SNARF [Chen et al. 2023] and COAP [Mihajlovic et al. 2022]. Additionally, we conducted ablation studies to assess the influence of the network architecture on reconstruction accuracy and model complexity.

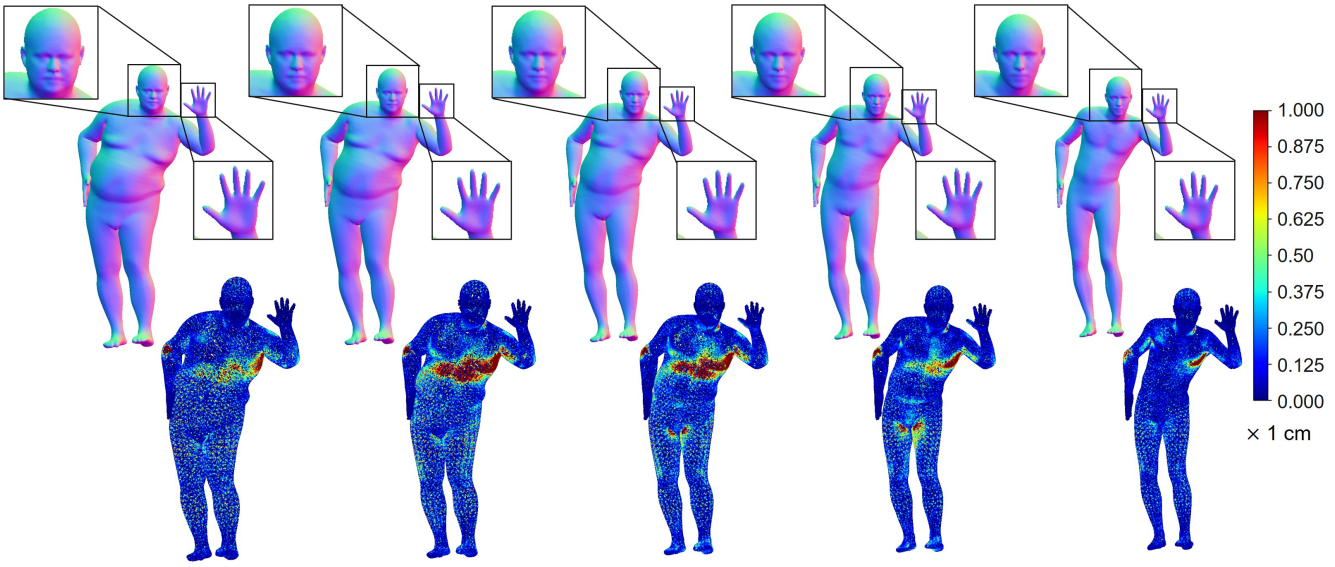


Figure 4: Top: Meshes obtained with our method. The three characters in the middle do not belong to the training set. They are obtained by conditioning the networks with interpolated SMPL identity parameters (β), between the far left and right characters. Zooms on the hand and the face show the ability to reproduce high frequency details. Bottom: the relative distance for each point to the corresponding SMPL [Loper et al. 2015] mesh.

Table 1: Comparison between Fast-SNARF, COAP, and our method. Three metrics: IoU bbox (the mean IoU of points sampled uniformly in space), IoU surface (the mean IoU of points sampled close to the surface) and Chamfer distance (the average distance of $20k$ points sampled on the output surface). Our model outperforms both methods in most metrics while using fewer parameters.

	Seen poses and identities			Unseen poses but seen Id			Unseen Id and poses			Number
	IoU Bbox (%) ↑	IoU Surf (%) ↑	Cham Dist (cm) ↓	IoU Bbox	IoU Surf	Cham Dist	IoU Bbox	IoU Surf	Cham Dist	of param (millions)
COAP (female)	94.26	94.23	0.49	92.57	90.36	0.51	89.77	88.45	0.55	0.84
Fast-SNARF	98.24	93.66	0.30	96.82	89.03	0.34	N/A	N/A	N/A	2.60
NIPIG (female)	**98.74**	**95.38**	**0.29**	**97.57**	**91.48**	**0.32**	**97.06**	**89.76**	**0.33**	0.52
NIPIG Light (female)	98.22	93.48	0.30	97.17	89.77	0.33	96.45	87.77	0.35	**0.13**
COAP (male)	93.43	93.10	0.54	92.18	90.45	0.56	90.50	**89.25**	0.57	0.84
Fast-SNARF	98.32	93.52	0.31	97.26	89.67	0.34	N/A	N/A	N/A	2.6
NIPIG (male)	**98.68**	**94.77**	**0.30**	**97.67**	**90.32**	**0.33**	**97.08**	88.82	**0.30**	0.52
NIPIG Light (male)	98.14	92.80	0.32	97.29	89.94	0.34	96.07	85.39	0.39	**0.13**
COAP (neutral)	95.45	**94.92**	0.48	93.83	**91.72**	0.50	N/A	N/A	N/A	0.84
Fast-SNARF	98.28	93.59	0.30	97.04	89.35	0.34	N/A	N/A	N/A	5.2
NIPIG (all)	**98.46**	94.30	**0.30**	**97.42**	90.65	**0.33**	96.20	86.35	0.37	0.52
NIPIG Light (all)	97.90	92.20	0.31	97.10	89.29	0.34	95.92	85.36	0.38	**0.13**

6.1 Qualitative evaluation

Figure 4 shows the interpolation between two known identities. We zoomed on challenging areas such as the face or the hand to highlight the smooth interpolation. The bottom row highlight the distance between each randomly sampled point on the generated mesh to its closest point on the ground truth SMPL [Loper et al. 2015] mesh. As expected, the interpolated shapes present a slightly higher error as they were unseen during the training. Figure 5 shows that the model works in challenging poses coming from the MPI PosePrior dataset [Akhter and Black 2015], even with unseen identities (see the videos in the supplementary materials for more results). SMPL [Loper et al. 2015] enables interpolation between identities but with the same gender, whereas our model can also interpolate between male and female characters, by conditioning the occupancy network with a gender code (see Figure 6). The model can also be conditioned on other parameters such as on the character's weight. We approximated each character's weight

before the training from the meshes' volume (see the supplementary materials for visual weight editing results). We then present a comparison with other SOTA methods: COAP [Mihajlovic et al. 2022] and Fast-SNARF [Chen et al. 2023]. COAP result is obtained with the pretrained weights given by the authors. The mesh has a visible artifact on the belly where two body parts joint. With our sampling strategy, compare to Fast-SNARF, NIPIG is more detailed on the face but less on other region like the belly button. Finally, we show how the model's size impact the visual quality. Figure 8 shows that a wider network mainly increases the high-frequency details, most visible on the face. However, non of the metrics we used are suitable for discriminating high frequencies and it remains an area of improvement. For instance, as presented in Table 6, increasing the occupancy network's width from 128 to 512 barely impacts the chamfer metric.

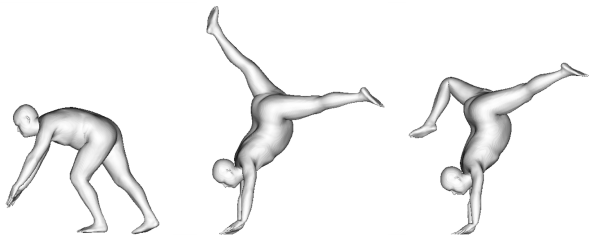


Figure 5: Example of extreme reposing far from the training set. The sequence is taken from the MPI PosePrior dataset [Akhter and Black 2015], and the identity was unseen during the training as it is interpolated between two known identities. There are no visible artifacts.

Figure 6: Example of identity and gender interpolation between two known characters (far left and far right). The shapes are smoothly interpolated

6.2 Quantitative evaluation

This section presents the accuracy of the reconstruction using our method, when the characters are within or out the training distribution. We also analyzed how the architecture of the networks impact this accuracy.

6.2.1 Within training distribution. Similarly to COAP, we used a single model for each experiment ('male', 'female', and 'all'). Across all experiments, our model obtained better results than Fast-SNARF

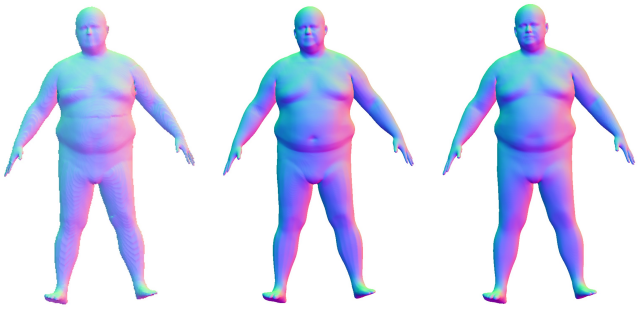


Figure 7: Comparison with SOTA methods. Left to right: COAP, Fast-SNARF, Ours. COAP has artifacts between body parts while Fast-SNARF lacks details on the face

Figure 8: Different occupancy network architectures with a width of, from left to right, 128, 256, and 512. They respectively have 134k, 522k and 1900k parameters

while being approximately n times lighter (n = number of subjects, see 'number of parameters' in Table 1). Although having similar Intersection over Union results compare to COAP, we have significantly better results on the Chamfer distance. Our model also has 38% fewer parameters than COAP. We hypothesize that the Fast-SNARF model was probably over-parameterized, thus allowing us to represent much more identities with the same number of parameters. The quantitative improvement could be either due to some training changes (the addition of a learning rate scheduler, the switch from Adam to AdamW, a longer training...), or from the more diverse dataset. On the other hand, the improvement compared to COAP might be explained by the difference of architecture: COAP uses several body-parts, which leads to visible artifacts.

Finally, we show the quantitative results obtained with the lighter model: although slightly lower than the base model, they also require much less parameters.

Results in Table 2 show that the performances of our model only slightly decrease, even for up to 38 characters in the training set. As a reminder, if not stated otherwise, all models have the same architecture as described in section 4.1.

6.2.2 Out of training distribution. We define out-of-distribution data as data that cannot be interpolated from the training set, thus measuring the extrapolation capabilities of our model. For both male and female datasets (each composed of five subjects), we trained the model on four subjects and tested it on the fifth. We used the SMPL

Table 2: Comparison of different dataset sizes. The model is able to represent up to 38 subjects with almost no degradation in terms of metrics. We also the results obtained with a learned latent code instead of using SMPL's β parameter.

	Seen poses and identities			Unseen poses but seen Id			Unseen Id and poses			Number
	IoU Bbox (%) ↑	IoU Surf (%) ↑	Cham Dist (cm) ↓	IoU Bbox	IoU Surf	Cham Dist	IoU Bbox	IoU Surf	Cham Dist	of param (millions)
10 subjects (default)	**98.46**	**94.21**	**0.303**	**97.45**	90.65	0.329	**96.20**	**86.35**	**0.375**	0.522
20 subjects	98.38	94.10	0.304	97.35	**90.69**	**0.329**	96.05	85.92	0.387	0.522
38 subjects	98.37	93.94	0.311	97.40	90.62	0.339	95.70	84.79	0.409	0.522
38 subjects (learned latent code)	98.38	93.92	N/A	97.39	90.54	N/A	N/A	N/A	N/A	0.522

Table 3: IoU surface and Chamfer distance (in cm) metrics computed with the fine-tuned model, either on the new character or characters from the training data (other characters). We compared the results obtained in the unsupervised setting (Unsuper), and the supervised setting (Super) and with the metrics obtained from Fast-SNARF as a baseline. The supervised setting significantly improved the overall performance for the "other characters" while slightly deteriorating the results for a "new character".

	New character		Other characters	
	IoU surf	Chamfer	IoU surf	Chamfer
Unsuper	94.10	0.30	52.29	1.35
Super	92.56	0.31	90.27	0.33
F-SNARF	93.59	0.30	N/A	N/A

Table 4: The table shows the Chamfer distance for the "fine-tuned" character and the "other characters". The first row shows results for fine-tuning the full model, the second row only the occupancy network, and last row the first 3 layers of the occupancy as well as the whole LBS network.

	Fine-tuned character	Other characters	Number of parameters
Full model	0.31	0.33	5.23E05
Occupancy	0.41	0.33	4.68E05
L0-3 and LBS	0.32	0.33	2.56E05

[Loper et al. 2015] β parameter of the unseen character as identity-conditioning input. We then fine-tuned the pre-trained model using the unseen character's training data. We used a BCE loss similarly to the original training and fine-tuned for 1000 iterations, with a batch size of 8. When unsupervised, Table 3 shows that for the fine-tuned character, we achieved similar results to Fast-SNARF [Chen et al. 2023], for both the IoU and the Chamfer distance. However, we noticed that only fine-tuning on one character degrades significantly the two metrics for the other characters (see Table 3 for the metrics and the supplementary material for visual results). As explained in 4.4, this is called catastrophic forgetting: the network loses previous knowledge. We use the teacher-student approach to mitigate this.

We noticed that in this setting, the quantitative results on the fine-tuned character only slightly worsen (see Table 3), while the metrics on the other characters dramatically improve. We have also tested to limit the number of parameters needed for the fine-tuning by freezing specific layers (see Table 4). Fine-tuning the first three layers of the occupancy and the entire LBS networks reduced the number of parameters by more than half. This adjustment barely increased the Chamfer distance of the fine-tuned character. (see Table 4). However, freezing the LBS network increased the Chamfer distance by 74%., with a marginal gain regarding the number of parameters. Thus indicating that the first layers of both networks play an important role to adapt to new identities.

6.2.3 Architecture study. To search for the optimal network architecture and size, we tuned the depth (number of layers) and/or the width (number of neurons per layer) of the network. The total number of combinations would be very high, and we decoupled the two parameters: tune the depth independently from the width, and reversely. As seen on Table 6, it is impossible to define the best model size: deeper and bigger networks have better performances but also requires more trainable parameters. The choice of the model depends on the use-case. In most of our experiments, we use the same architecture as Fast-SNARF for a more fair comparison: 8 fully connected layers of size 256 for the occupancy network, and 4 fully connected layers of size 128 for the LBS network. Another important choice is the layer at which the identity and pose codes are introduced in the networks. Our experimental results (see Table 6) showed that the earlier the model is conditioned on the identity, the better. Indeed, the identity is a global and low frequency feature that needs to impact the entire network. On the other hand, the pose conditioning can happen at the end of the network as it only refines the final shape. Hence, the optimal choice seems to condition the networks at the first layer. Finally, when comparing between three popular conditioning methods (Table 5), the gain provided by the more complex hypernetwork is negligible compared to the induce increased number of parameters. On the other hand, FiLM tends to struggle to generalize. We therefore chose the simplest, but yet more versatile conditioning method : the concatenation.

7 CONCLUSION

We propose a lightweight neural implicit generative model for avatars. By being carefully conditioned on an identity parameter, our model is capable of representing a wide set of shapes while only being trained on a few subjects. This approach enables efficient and flexible avatar generation, making it suitable for various applications. We demonstrate that training on multiple identities can also improve the model's results on single identities. This training strategy not only improves individual identity representation but

Table 5: Different conditioning methods. The hypernetwork achieve the best quantitative results but at the cost of doubling the number of parameters. FiLM and concatenation are similar, except for unseen identities where FiLM degrades significantly.

	Seen poses and identities			Unseen poses but seen Id			Unseen Id and poses			Number
	IoU Bbox (%) ↑	IoU Surf (%) ↑	Cham Dist (cm) ↓	IoU Bbox	IoU Surf	Cham Dist	IoU Bbox	IoU Surf	Cham Dist	of param (millions)
Hypernetwork	**98.90**	**95.62**	**0.301**	**97.75**	**91.59**	**0.329**	96.99	**88.92**	**0.352**	1.15
FiLM	98.51	94.29	0.305	97.63	91.20	0.329	92.52	77.23	0.612	0.526
Concatenation	98.60	94.70	0.303	97.58	91.08	0.330	**96.70**	88.08	0.357	**0.522**

Table 6: The table shows the Chamfer distance for different occupancy model architectures applied to the training dataset, unseen poses, and unseen poses and identities. The table is divided row-wise in different experiments, firstly, results for various numbers of layers; secondly, results for the dimensions of each layer; lastly, the indexes of the conditioned layers on the identity parameter. The values in bold represent the best results in each experiment.

	Chamfer distance (cm) ↓					
# layers occ	4	6	8	10	15	
Train	0.33	0.32	**0.30**	0.31	0.31	
Pose out	0.36	0.35	**0.33**	0.34	0.33	
ID out	0.39	0.39	**0.37**	0.40	0.42	
Size layers	32	64	128	256	512	
Train	0.37	0.33	0.31	0.30	**0.30**	
Pose out	0.39	0.35	0.34	**0.33**	0.33	
ID out	0.53	0.43	0.39	0.37	**0.37**	
Id cond layer	0	1	2	3	5	7
Train	0.31	0.31	**0.30**	0.31	0.31	0.31
Pose out	0.34	**0.33**	0.33	0.33	0.33	0.33
ID out	**0.36**	0.37	0.39	0.40	0.40	0.65

also makes increases generalization, without increasing the computational cost. We thoroughly illustrate its interpolation capabilities using gender and identity conditioning. Additionally, we highlight some limitations for extrapolation, particularly for morphologies far from the training set distribution. Although we could still obtain plausible shapes, the result inevitably deviate from ground truth SMPL's mesh as we steer away from known identities. However, thanks to the knowledge acquired by the model while training on other identities, a few iterations of fine-tuning are enough to generate results similar to those obtained with the default training. Furthermore, by carefully selecting the parts of the network to fine-tune, we can reduce the number of trainable parameters by about half. Our experiments demonstrate that larger networks do not necessarily yield better quantitative results, but can be visually impactful, especially for high-frequency details. Furthermore, the placement of identity conditioning within the network is crucial for its ability to generalize to new identities. We believe that studying and presenting results for different model architecture help find the best compromise between quality and size. Overall, reducing the training and storage cost of AI models becomes a vital issue to tackle, especially for massively multi-user applications with personalized avatars.

ACKNOWLEDGMENTS

This work was partly supported by Inria through the Ys.AI and Nemo.AI projects, and by a CIFRE PhD fellowship (No. 2023/0308) funded by ANRT in partnership with InterDigital.

REFERENCES

Ijaz Akhter and Michael J. Black. 2015. Pose-Conditioned Joint Angle Limits for 3D Human Pose Reconstruction. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) 2015*.

Federica Bogo, Javier Romero, Gerard Pons-Moll, and Michael J. Black. 2017. Dynamic FAUST: Registering Human Bodies in Motion. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Honghu Chen, Yuxin Yao, and Juyong Zhang. 2025. Neural-ABC: Neural Parametric Models for Articulated Body With Clothes. *IEEE Transactions on Visualization and Computer Graphics* 31, 2 (2025), 1478–1495. https://doi.org/10.1109/TVCG.2024.3364814

Xu Chen, Tianjian Jiang, Jie Song, Max Rietmann, Andreas Geiger, Michael J. Black, and Otmar Hilliges. 2023. Fast-SNARF: A Fast Deformer for Articulated Neural Fields. *Pattern Analysis and Machine Intelligence (PAMI)* (2023).

Xu Chen, Tianjian Jiang, Jie Song, Jinlong Yang, Michael J Black, Andreas Geiger, and Otmar Hilliges. 2022. gDNA: Towards Generative Detailed Neural Avatars. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Xu Chen, Yufeng Zheng, Michael J Black, Otmar Hilliges, and Andreas Geiger. 2021. SNARF: Differentiable Forward Skinning for Animating Non-Rigid Neural Implicit Shapes. In *International Conference on Computer Vision (ICCV)*.

Enric Corona, Albert Pumarola, Guillem Alenyà, Gerard Pons-Moll, and Francesc Moreno-Noguer. 2021. SMPLicit: Topology-aware Generative Model for Clothed People. In *CVPR*.

Boyang Deng, JP Lewis, Timothy Jeruzalski, Gerard Pons-Moll, Geoffrey Hinton, Mohammad Norouzi, and Andrea Tagliasacchi. 2020. Neural Articulated Shape Approximation. In *The European Conference on Computer Vision (ECCV)*. Springer.

Zijian Dong, Chen Guo, Jie Song, Xu Chen, Andreas Geiger, and Otmar Hilliges. 2022. PINA: Learning a Personalized Implicit Neural Avatar from a Single RGB-D Video Sequence. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), 20438–20448. https://api.semanticscholar.org/CorpusID:247222837

Yao Feng, Haiwen Feng, Michael J. Black, and Timo Bolkart. 2021. Learning an animatable detailed 3D face model from in-the-wild images. *ACM Trans. Graph.* 40, 4, Article 88 (July 2021), 13 pages. https://doi.org/10.1145/3450626.3459936

Meiying Gu, Jiahe Li, Yuchen Wu, Haonan Luo, Jin Zheng, and Xiao Bai. 2025. 3D human avatar reconstruction with neural fields: A recent survey. *Image and Vision Computing* 154 (2025), 105341. https://doi.org/10.1016/j.imavis.2024.105341

David Ha, Andrew M. Dai, and Quoc V. Le. 2016. HyperNetworks. *CoRR* abs/1609.09106 (Sept. 2016). https://doi.org/10.48550/ARXIV.1609.09106 arXiv:1609.09106 [cs.LG]

David A. Hirshberg, Matthew Loper, Eric Rachlin, and Michael J. Black. 2012. Coregistration: Simultaneous Alignment and Modeling of Articulated 3D Shape. In *European Conference on Computer Vision*. https://api.semanticscholar.org/CorpusID:19014218

Zexu Huang, Sarah Monazam Erfani, Siying Lu, and Mingming Gong. 2024. Efficient neural implicit representation for 3D human reconstruction. *Pattern Recognition* 156 (2024), 110758. https://doi.org/10.1016/j.patcog.2024.110758

Ladislav Kavan, Steven Collins, Jiří Žára, and Carol O'Sullivan. 2007. Skinning with dual quaternions. In *Proceedings of the 2007 Symposium on Interactive 3D Graphics and Games* (Seattle, Washington) *(I3D '07)*. Association for Computing Machinery, New York, NY, USA, 39–46. https://doi.org/10.1145/1230100.1230107

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Trans. Graph.* (2023).

Mengtian Li, Shengxiang Yao, Zhifeng Xie, and Keyu Chen. 2024. GaussianBody: Clothed Human Reconstruction via 3d Gaussian Splatting.

Ruilong Li, Shan Yang, David A. Ross, and Angjoo Kanazawa. 2021. Learn to Dance with AIST++: Music Conditioned 3D Dance Generation. arXiv:2101.08779 [cs.CV]

Tianye Li, Timo Bolkart, Michael. J. Black, Hao Li, and Javier Romero. 2017. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)* 36, 6 (2017), 194:1–194:17. https://doi.org/10.1145/3130800.3130813

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. 2015. SMPL: A Skinned Multi-Person Linear Model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)* 34, 6 (Oct. 2015), 248:1–248:16.

Lars M. Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. 2018. Occupancy Networks: Learning 3D Reconstruction in Function Space. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2018), 4455–4465. https://doi.org/10.1109/cvpr.2019.00459

Marko Mihajlovic, Shunsuke Saito, Aayush Bansal, Michael Zollhoefer, and Siyu Tang. 2022. COAP: Compositional Articulated Occupancy of People. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. IEEE. https://doi.org/10.1109/cvpr52688.2022.01285

Marko Mihajlovic, Yan Zhang, Michael J Black, and Siyu Tang. 2021. LEAP: Learning Articulated Occupancy of People. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. IEEE. https://doi.org/10.1109/cvpr46437.2021.01032

Gyeongsik Moon, Takaaki Shiratori, and Shunsuke Saito. 2024. Expressive Whole-Body 3D Gaussian Avatar. In *ECCV*.

Jeong Joon Park, Peter R. Florence, Julian Straub, Richard A. Newcombe, and S. Lovegrove. 2019. DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2019), 165–174. https://doi.org/10.1109/cvpr.2019.00025

Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. 2019. Expressive Body Capture: 3D Hands, Face, and Body from a Single Image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron C. Courville. 2017. FiLM: Visual Reasoning with a General Conditioning Layer. *Proceedings of the AAAI Conference on Artificial Intelligence* abs/1709.07871, 1 (April 2017). https://doi.org/10.1609/aaai.v32i1.11671 arXiv:1709.07871 [cs.CV]

Marco Pesavento, Yuanlu Xu, Nikolaos Sarafianos, Robert Maier, Ziteng Wang, Chun-Han Yao, Marco Volino, Edmond Boyer, Adrian Hilton, and Tony Tung. 2024. ANIM: Accurate Neural Implicit Model for Human Reconstruction from a Single RGB-D Image. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024), 5448–5458.

Shunsuke Saito, Jinlong Yang, Qianli Ma, and Michael J. Black. 2021. SCANimate: Weakly Supervised Learning of Skinned Clothed Avatar Networks. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*. IEEE. https://doi.org/10.1109/cvpr46437.2021.00291

Garvita Tiwari, Nikolaos Sarafianos, Tony Tung, and Gerard Pons-Moll. 2021. Neural-GIF: Neural Generalized Implicit Functions for Animating People in Clothing. In *International Conference on Computer Vision (ICCV)*.

Shaofei Wang, Marko Mihajlovic, Qianli Ma, Andreas Geiger, and Siyu Tang. 2021. Metaavatar: Learning animatable clothed human models from few depth images. 34 (2021), 2810–2822.

Heming Zhu, Fangneng Zhan, Christian Theobalt, and Marc Habermann. 2024. TriHuman: A Real-time and Controllable Tri-plane Representation for Detailed Human Geometry and Appearance Synthesis. *ACM Trans. Graph.* 44, 1, Article 4 (Oct. 2024), 17 pages. https://doi.org/10.1145/3697140

8 EXTENDED DATASET

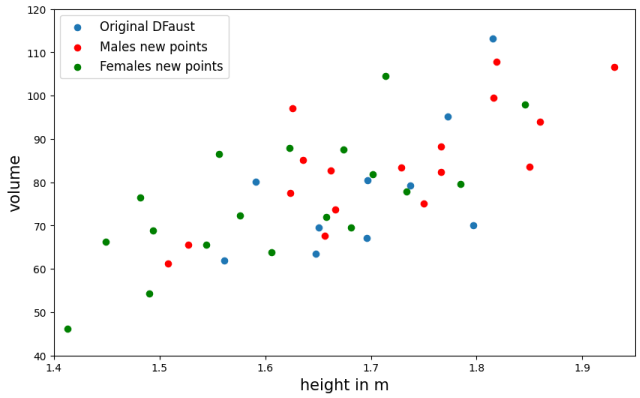


Figure 9: Plot of all characters in the extended dataset. In the x-axis is the height in meters, and in the y-axis is the volume of the mesh, rescaled to approximate a weight in kg. We can notice that our extended dataset has more diverse and extremes shapes compared to DFaust.

Figure 9 shows a 2D plot of all 38 characters composing the extended synthetic dataset presented in section 5.1. To be more intuitive, the volume is rescaled to better approximate the weight. We can notice that the green and red dots representing the newly created identities are more diverse than the original DFaust characters in blue.

9 ARCHITECTURE STUDY

In the following section, we showcase more detailed results from the main document. We measured the Chamfer distance for all three datasets (training data, unseen poses and unseen poses + identities) for different architecture variation (figures 10, 11, 12). For all three figures, we only present the Chamfer distance to avoid redundancy, as we observed similar behavior for the IoU surface and IoU uniform metrics. Figure 10 shows the variation of the Chamfer distance with respect to the number of layers in the occupancy network. As we want to find a good trade-off between size and quality, we added the evolution of the number of parameters. In figure 11, we plotted the Chamfer with respect to the size of each layer in the occupancy network. Finally, in figure 12, we plotted the Chamfer distance with respect to the index of the layer conditioned on the identity parameter. We observe similar conclusion as in the main document: it is detrimental to use more than 8 layers and of size more than 256 regarding the generalization capabilities for unseen identities. Furthermore, we see an improvement for unseen identities when conditioning the network in early layers.

9.1 Fine-tuning

Here, we give more details about the parameters needed to fine-tune the model on a new identity. First, before fine-tuning the weights of the model, we tried to optimize the value of the SMPL [Loper et al. 2015] identity parameter β. As we condition our network on this parameter, we expect our identity space to be aligned with the

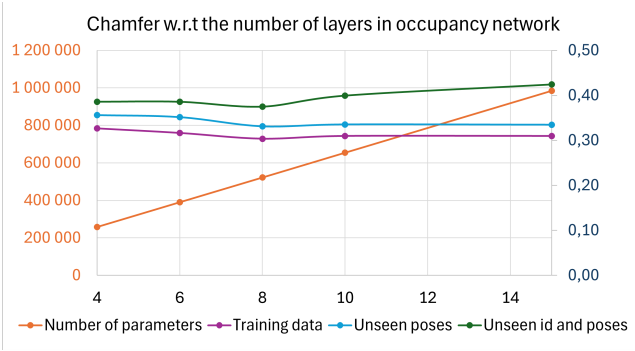


Figure 10: Chamfer distance computed on "training data", "unseen poses" and "unseen poses + identities", with respect to the number of layers in the occupancy network.

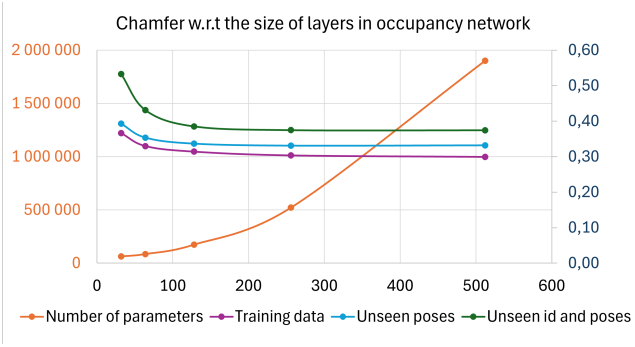


Figure 11: Chamfer distance computed on in-distribution, pose out-of-distribution and finally pose+identity out-of-distribution, with respect to the size of layers in the occupancy network.

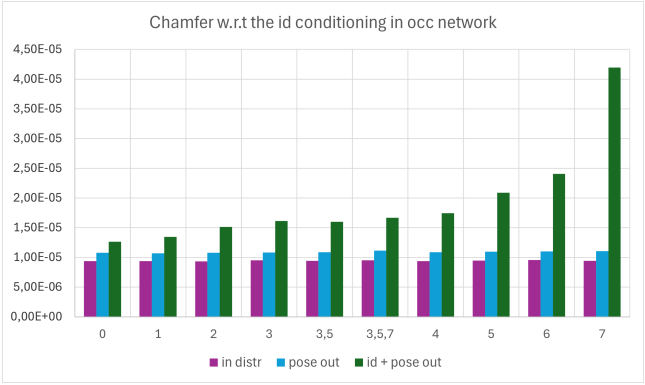
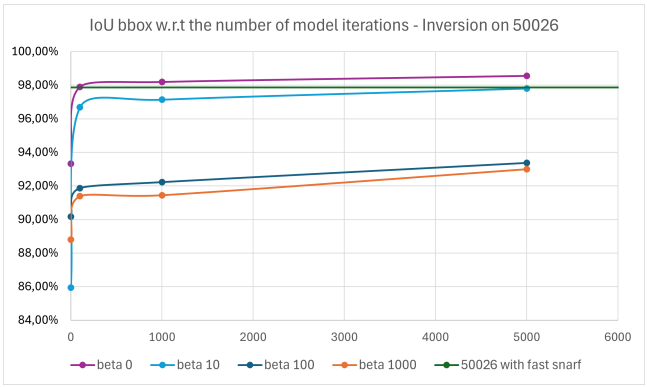
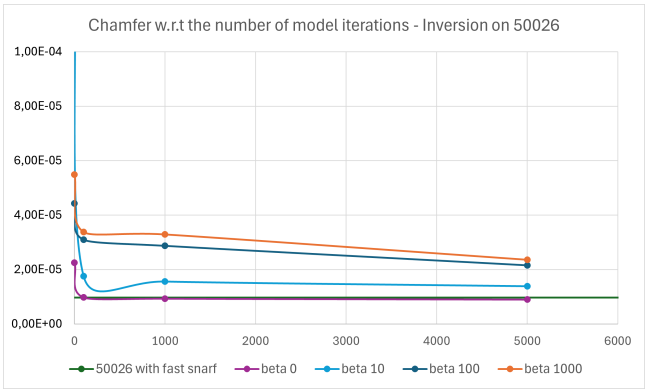


Figure 12: Chamfer distance computed on in-distribution, pose OOD and finally pose+identity OOD, with respect to identity-conditioned layers in the occupancy network.

one of SMPL, for identities used during the training. However, we could be representing only a subset of SMPL's identity space and can not be certain that it extrapolates to any new identity. Thus,



(a)



(b)

Figure 13: Plot of the different metrics (Chamfer distance 13b, and IoU for uniform points 13a) computed on a new character (here 50026) not seen during the training, the different beta curves correspond to the number of iterations to optimize the beta parameter, the x-axis coordinates correspond to the number of iterations to fine-tune the model's weights

we started from the SMPL [Loper et al. 2015] β identity parameter of the new character and optimized it's value for a few iterations. We then froze the identity parameter and fine-tune the model's weights. As seen in figure 13, this approach was detrimental ; the more we optimized the β, the worst were the fine-tuning later. This result confirmed that are model is well aligned with the SMPL's identity latent space. In figure 13, we show that we managed to achieve even better results than Fast SNARF [Chen et al. 2023] with only 100 iterations of fine-tuning. Although it is not the case for every character, the model achieves on average similar results as Fast SNARF [Chen et al. 2023].

The figure 14 shows the difference between supervising or not the fine-tuning. During the first few iterations, in both settings, the accuracy of the model for the other characters (seen during the original training) drops. However, by supervising the model to also copy previous predictions, as explained in the main document, the accuracy quickly goes back up. Regarding the fine-tuned character, although the training converge less rapidly, both settings achieve the same accuracy. Thus, the supervision could mitigate the catastrophic forgetting effect with no drawbacks.

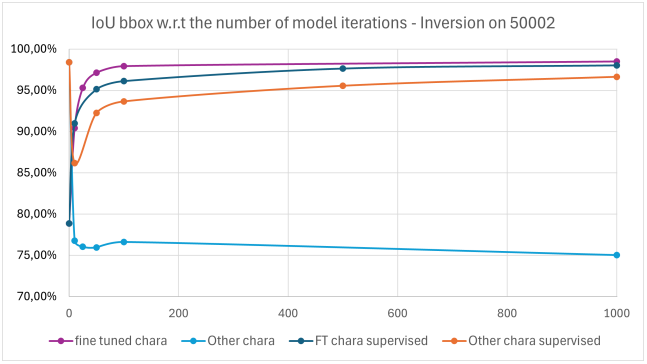


Figure 14: Plot of the IoU for uniform points, computed on a new fine-tuned character (here 50002) not seen during training, and on previous identities ('other chara') seen during the training. We see that without supervision, the model correctly overfit on the new identity but quickly degrades for the other identities. However, by supervising the model on old predictions, we can achieve similar results for the fine-tuned character while mitigating the dropout on the other identities.

The Chamfer distance is great to compare our overall shape to a ground truth. However, we noticed that it was not so sensible to local artifacts or high frequency details. Similarly, supervising the fine-tuning as explained in the main document, seems to greatly improve the quantitative results for the other characters (the one used during the original training). Yet, in some cases, the visual difference is barely noticeable (see figure 15). As future work, we thus would be interested in conducting a perceptual study to better assess the visual quality of our results.

9.2 Ablation study

Table 7 compare the different metrics when we do not condition the occupancy or the LBS network on the identity. We can see that we obtain the best results by conditioning both network on the identity. Not conditioning the occupancy network yields much worsts results while not conditioning the LBS network seems to have a light impact.

Table 7: Ablation study for unseen poses and identities, when either the occupancy or the LBS network is not conditioned on the identity parameter. Conditioning the occupancy network on the identity is much more important than conditioning the LBS network.

Method	Chamf distance	IoU bbox	IoU surf
w/o occupancy cond on ID	2.57E-4	78.50	52.31
w/o LBS condition on ID	1.39E-05	96.59	87.72
Full	**1.28E-05**	**96.69**	**88.05**

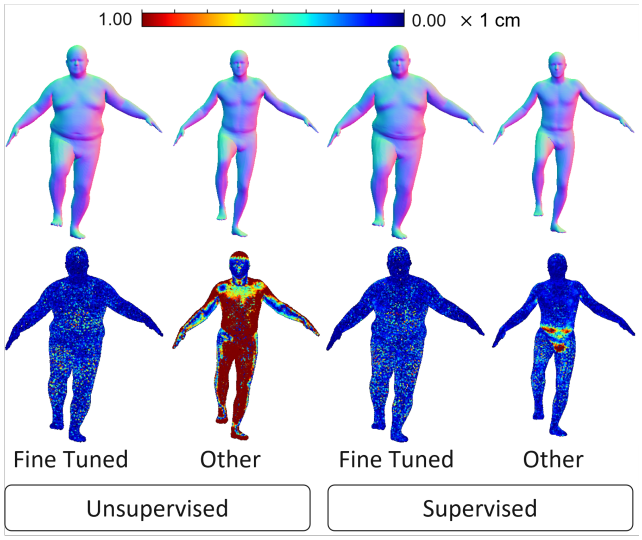


Figure 15: Visual comparison when the fine-tuning is supervised or unsupervised. We see that while the fine-tuned character in the supervised setting has a bigger Chamfer distance (see Table 3), it is visually very similar to the unsupervised one. The difference is much more pronounced for the other character by looking at the Chamfer distance (bottom row).

9.3 Identity interpolation

We present more qualitative results about the identity and gender interpolation (see figure 16)

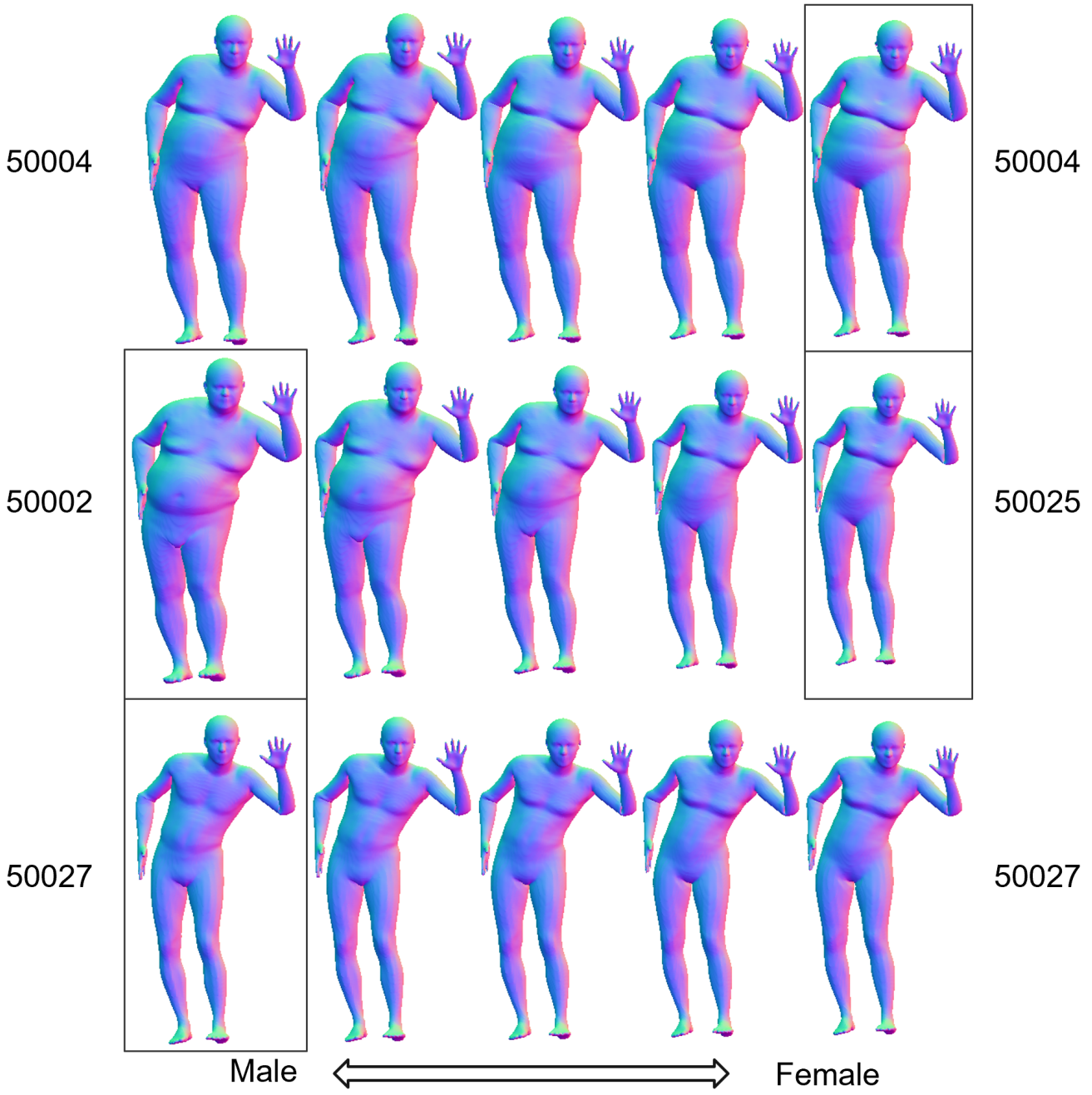


Figure 16: Visual results of the gender interpolation. The first and third rows represents a character with a fix identity β code (respectively character 50004 and 50027 in the DFaust dataset [Bogo et al. 2017]) while we interpolate the gender latent code between 0 and 1. The second row shows the interpolation between the characters 50002 and 50025 and thus requires to interpolate both the identity code β and the gender code. The black boxes represents the identities seen during the training. All examples show a smooth interpolation.