Reconstructing Human Body Mesh from Point Clouds by Adversarial GP Network

Boyao Zhou, Jean-Sébastien Franco, Federica Bogo, Bugra Tekin, Edmond Boyer

Reconstructing Human Body Mesh from Point Clouds by Adversarial GP Network

Boyao Zhou[1⋆], Jean-Sébastien Franco[1], Federica Bogo[2], Bugra Tekin[2], and Edmond Boyer[1]

[1] Inria-Univ. Grenoble Alpes-CNRS-Grenoble INP-LJK, France
{boyao.zhou, jean-sebastien.franco, edmond.boyer}@inria.fr
[2] Microsoft Zürich, Switzerland
{febogo,bugra.Tekin}@microsoft.com

Abstract. We study the problem of reconstructing the template-aligned mesh for human body estimation from unstructured point cloud data. Recently proposed approaches for shape matching that rely on Deep Neural Networks (DNNs) achieve state-of-the-art results with generic pointwise architectures; but in doing so, they exploit much weaker human body shape and surface priors with respect to methods that explicitly model the body surface with 3D templates. We investigate the impact of adding back such stronger shape priors by proposing a novel dedicated human template matching process, which relies on a point-based, deep autoencoder architecture. We encode surface smoothness and shape coherence with a specialized Gaussian Process layer. Furthermore, we enforce global consistency and improve the generalization capabilities of the model by introducing an adversarial training phase. The choice of these elements is grounded on an extensive analysis of DNNs failure modes in widely used datasets like SURREAL and FAUST. We validate and evaluate the impact of our novel components on these datasets, showing a quantitative improvement over state-of-the-art DNN-based methods, and qualitatively better results.

1 Introduction

Template-based human shape matching is a problem of broad interest in computer vision, for a variety of applications relevant to Augmented and Virtual Reality, surveillance and 3D media content production. It is relevant to various tasks such as dense shape alignment or tracking, shape estimation and completion from sparse or corrupt shape data.

This problem has been addressed with several classic approaches that either directly find dense correspondence using intrinsic surface embeddings [1–3] or use human body templates as geometric proxy to guide the matching [4–8]. Both approaches usually involve some form of non-convex optimization that

⋆ MSR-INRIA Joint Center

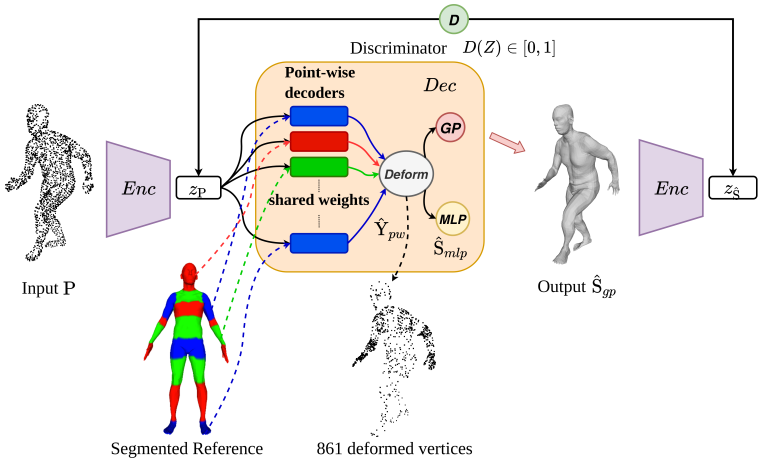


Fig. 1. Overview of our Adversarial GP Network.

is susceptible to ambiguities and local minima, and hand-crafted features to estimate the correspondence.

Aiming for noise and initialization resilience and improvement in feature description has motivated an avenue of research in learning-based correspondence approaches to the human shape matching problem. These methods have the property to automate feature extraction and matching by mining large datasets, and can estimate correspondences by building automatic feature classifiers with e.g. random forests [9], or simultaneously learn feature extraction and correspondence using DNNs [10–14].

Many of these learning approaches rely on some form of human a priori knowledge. Most methods propose matching to an explicit shape deformation model, for which a reduced parameterization is predicted [15–17] or whose mapping to the data is learned [18, 19] from observations.

Among the most successful approaches of inspiration to this work are those matching human shapes using an implicit deformation model which is entirely learned with no manually set components, as applied to humans [20] or generic objects [21]. By encoding matching to an underlying template as the expression of a learned global feature in a latent space automatically discovered by an auto-encoder, the model can be entirely automated and trained end-to-end for generic matching of two shapes, as opposed to the previously described methods. As they use point-based DNN architectures [22], these approaches can be applied to point cloud inputs without any surface consistency. All these properties afford greater robustness and generalization abilities, and allow this family of methods to outperform the latter on standard benchmarks. However in this process a weaker human shape and consistent surface prior is encoded than previous approaches, which leads to noisy, and sometimes non-realistic predicted human shapes, as confirmed by an analysis of the failure cases of these approaches.

Some of the failures are mitigated using a post-processing step which consists in optimizing the shape matching features inferred by the DNN in the latent space, which improves the final result.

In this work, we explore within this family of approaches how local and global shape priors, commonly not encoded with point-wise architecture, can be reintroduced while maintaining the benefits of such an architecture (e.g. PointNet [22]). We base our approach on a point-based auto-encoder similar to [20], but with several key differences. To alleviate inference noise, we introduce a Gaussian Process decoder layer which inherently encodes surface smoothness and surface point coherence on the shape with lower point dimensionality on the surface, only to the price of a small pretraining phase. Second, more global consistency is built in the model by adding fully connected layers at the end of the decoder, which is made possible by the surface dimensionality reduction previously discussed. Third, to avoid inferring drastically non human shapes, we introduce an adversarial training phase inspired by [23] which enforces consistency of human shape encodings in our latent space and helps to avoid overfitting. With these improved network characteristics and training procedures, we show that our approach provides results that are on par or better than state-of-the-art on the FAUST intra and inter challenges and illustrate the quality gain of our approach through an exhaustive ablation study illustrating the benefits of these three contributions.

2 Related Work

There exists a rich literature on registration and reconstruction of 3D data (see [24, 25] for a survey). Here, we focus our analysis on methods for registering human body shapes, following the classic distinction between template-free and template-based methods. We then briefly discuss how Gaussian Processes have been combined with DNNs in previous work, and the use of adversarial training in the context of 3D vision.

Template-free methods Correspondences between non-rigid objects can be established by defining an intrinsic surface representation, which is invariant to bending. In the embedding space defined by this representation, the registration problem boils down to a non-convex optimization one. Examples of intrinsic representations are Generalized Multi-Dimensional Scaling (GMDS) [1], heat kernel maps [2], Möbius transformations [3]. Recent work tries to learn such representations, and therefore object-to-object correspondences, from data. While early approaches rely on random forests [9], subsequent ones employ DNNs [10, 11]. For example, Deep Functional Maps [12, 13] combine a deep architecture with point-wise descriptors [14] to obtain dense correspondences between pairs of shapes. These methods aim at matching arbitrary shapes. However, when focusing on particular instances like the human body, one can introduce more powerful class-specific shape priors.

Template-based methods When registering noisy and incomplete 3D human body data, one commonly relies on a predefined 3D body template acting as a strong shape prior. At registration time, the template surface is deformed in order to match the data. Many approaches rely on a statistical body model [4, 5] and define an objective function which is minimized via non-linear least squares [6–8]. However, these objective functions use hand-crafted error terms and are not as powerful as data-driven approaches. Recently, the wider availability of huge datasets of 3D body shapes [7, 26] fostered the development of DNN-based methods. Mesh Variational Autoencoders [15] learn a latent space for 3D human body representation, but their input is limited to fixed-topology shapes. LBS-AE [16] proposes a self-supervised approach for fitting 3D models to point cloud. The method relies on DNNs to learn a set of Linear Blending Skinning [27] parameters. FARM [18] establishes correspondences between shapes by automatically extracting a set of landmarks and then using functional maps. Deep Hierarchical Networks [17] learn a 3D human body embedding which can then be fitted to data, leveraging a set of manually selected landmarks. Basis Point Sets [19] propose an efficient point cloud encoding, which can then be combined with DNNs [28] for shape registration and completion tasks.

GP and DNNs Gaussian Processes (GP) are popular in statistical learning for their generalization capabilities. In 3D vision, Lüthi *et al.* [29] propose GPMMs, a morphable model based on GP, with applications to face modeling and medical image analysis. Recently, some studies [30, 31] try to interpret how DNNs can simulate the learning process of GP. For example, Deep GP [32] focuses on probabilistic modeling of GP with DNNs, training the network via marginal likelihood. In this work, we leverage the interpolation and smoothness capabilities of GP in the context of 3D surface reconstruction.

Adversarial training After the introduction of Generative Adversarial Networks (GANs) [33], adversarial training has been widely used in computer vision. In 3D vision, HMR [34] applies adversarial learning to estimate 3D human body shape and pose from 2D images. CAPE [35] uses it to learn a model of people in clothing. Fernández Abrevaya *et al.* [36] and Shamai *et al.* [37] use adversarial training to model faces in 3D. Hu *et al.* [23] compare adversarial and L2-norm regularization for the task of image registration. To the best of our knowledge, our work is the first to propose adversarial training as a regularization term in the context of 3D registration.

In general, our work builds on 3D-CODED [20], which uses a PointNet-like [22] architecture to extract permutation-invariant point features. However it applies the point-wise decoders which are independent of each other. Thus we propose to strengthen the relationship of nearby points by using our GP layer and MLP layers. AtlasNet2 [38] aims at improving upon 3D-CODED reconstructions by using a learnable template. However AtlasNet2 results exhibit artifacts similar to the ones of 3D-CODED in some challenging cases. In order to make the network predictions more robust, we propose to use adversarial training.

3 Method

Our approach takes as input an unordered set of n 3D points $\mathrm{P} \in \mathbb{R}^{n \times 3}$ and maps this set into a deformed instance $\mathrm{S} \in \mathbb{R}^{res \times 3}$ of a reference mesh with a fixed resolution res. The number n of input points can vary. This to allow for partial or incomplete shape description as typical with laser scan or depth data. In order to learn such a mapping we use a point-wise encoder-decoder architecture trained on standard human body datasets. This architecture presents two innovations to better enforce shape consistency: first a regularization layer that builds on Gaussian Process (GP) and second a global adversarial loss. The sections below detail the different components of our framework.

3.1 Network Architecture

As shown in Figure 1, our architecture encodes points P into a latent shape representation $\mathcal{Z}_p \in \mathbb{R}^{1024}$ which is then decoded into a deformation vector field Y defined over a mesh template to produce the shape S. Our objective is to balance global and local information with shape-wise and point-wise considerations. To this end, a PointNet [22] like encoder is used as a backbone to extract the 1024-dimension latent shape feature $\mathcal{Z}_p$. On the decoder side, we first expect this global shape feature $\mathcal{Z}_p$ to predict the deformation of a subset of representative points on the reference mesh in a point-wise manner. More global considerations are then applied on this subset of points with both Gaussian Process interpolation and fully connected layers. Furthermore, to better constrain the latent representation during training, the output vertices of the predicted deformed reference mesh S are fed into the encoder to verify whether they yield a latent feature $\mathcal{Z}_s$ close to the latent feature $\mathcal{Z}_p$ of the ground truth shape vertices.

Encoder We extract the global feature $\mathcal{Z}_p$ with a simplified version of PointNet [22]. The input points P are first processed by 3 hidden layers of size 64, 128 and 1024, respectively, followed by a max-pooling operator applied to the resulting point-wise features. Then, two linear layers of size 1024 lead to the latent space $\mathcal{Z}_\mathcal{P}$. All layers use batch normalization and $ReLU$ (rectified linear unit) activation.

Decoder The decoder takes as input the shape feature $\mathcal{Z}_p$ extracted by the encoder together with l 3D locations x_i of vertices distributed on the reference mesh. Point-wise decoders with shared weights are first used on the combinations $(\mathrm{x}_i, \mathcal{Z}_p)$. These decoders are composed of 3 hidden layers going from size 1027 to 513 and 256. The resulting features are projected into l individual vertex deformations y_i using 2 times hyperbolic tangent activation functions. Following point-wise decoders, two computation flows are applied in parallel on the resulting predicted vertex deformations y_i. One goes to GP layers that enforce local spatial consistency between vertices and the other goes to a fully-connected MLP layer that enforces a global constraint over vertices. We take the output of the GP flow as the final deformed instance.

3.2 Local and Global Spatial Consistency

Gaussian Process Interpolation As mentioned before the decoder part includes a vertex interpolation technique based on Gaussian Process [39]. To this aim, we assume here that deformations y_i of the reference mesh at vertex locations x_i are, up to a bias $\varepsilon \sim \mathcal{N}(0, \sigma)$, non linear functions $\mathrm{y}_i = f(\mathrm{x}_i) + \varepsilon$, which distributions are jointly Gaussian, with mean and covariance defined by the kernel k:

$$k(\mathrm{x}_i, \mathrm{x}_j) = \gamma \exp(-\frac{\|\mathrm{x}_i - \mathrm{x}_j\|^2}{r}). \tag{1}$$

Under these assumptions, the joint distribution of l partial vertex observations Y and an unobserved vertex y_* over the deformed reference mesh can be expressed as:

$$\begin{bmatrix} \mathrm{Y} \\ \mathrm{y}_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} \mathrm{K}(\mathrm{X},\mathrm{X}) + \sigma^2 I, & \mathrm{K}_*^T(\mathrm{x}_*, \mathrm{X}) \\ \mathrm{K}_*(\mathrm{x}_*, \mathrm{X}), & k(\mathrm{x}_*, \mathrm{x}_*), \end{bmatrix}\right) \tag{2}$$

where $\mathrm{K}(,)$ denotes the covariances over the associated vertices x_i on the reference mesh:

$$\mathrm{K}(\mathrm{X},\mathrm{X}) = \begin{bmatrix} k(\mathrm{x}_1,\mathrm{x}_1) & \dots & k(\mathrm{x}_1,\mathrm{x}_l) \\ \vdots & \ddots & \vdots \\ k(\mathrm{x}_l,\mathrm{x}_1) & \dots & k(\mathrm{x}_l,\mathrm{x}_l) \end{bmatrix}, \ \mathrm{K}_*(\mathrm{x}_*,\mathrm{X}) = \begin{bmatrix} k(\mathrm{x}_*,\mathrm{x}_1) & \dots & k(\mathrm{x}_*,\mathrm{x}_l) \end{bmatrix}. \tag{3}$$

The posterior probability $P(\mathrm{y}_*|\mathrm{Y})$ can be inferred as a Gaussian distribution $\mathcal{N}(m(\mathrm{y}_*), var(\mathrm{y}_*))$ with:

$$m(\mathrm{y}_*) = K_*(K + \sigma^2 I)^{-1}\mathrm{Y} \tag{4}$$

$$var(\mathrm{y}_*) = k_* - K_*(K + \sigma^2 I)^{-1}K_*^T \tag{5}$$

where, to simplify our notation, $K_* = \mathrm{K}(\mathrm{x}_*, \mathrm{X})$, $K = \mathrm{K}(\mathrm{X},\mathrm{X})$ and $k_* = k(\mathrm{x}_*, \mathrm{x}_*)$. Taking the mean of this distribution as the predicted value we finally get:

$$\mathrm{y}_* = K_*(K + \sigma^2 I)^{-1}\mathrm{Y}. \tag{6}$$

In practice, to accelerate the GP computation and improve the reconstruction precision, we apply the above statistical reasoning individually over body parts instead of the full body. We follow for that purpose [40] and segment the body topology into 19 small patches, including two patches on the elbow (see Fig. 1). In addition, we do not consider absolute vertex locations as y_i but relative displacements with respect to the reference mesh instead. Note that we have finally 3 parameters: γ, r and σ for each body part. Thus we can use cross validation, more particularly in our case, *kernel selection*, to tune the GP parameters before the time-consuming gradient descent optimization during the neural network training.

In addition, the selected subset of l observation vertices impacts the final reconstruction of the full mesh. In order to select the most informative vertices for that purpose, we pre-tune 19 kernels and select the observation vertices using

10 random meshes from the FAUST training dataset. We start at 10% resolution of the template, i.e. 689 vertices, and progressively add vertices to minimize the reconstruction error, finding an optimal value of 861 vertices.

Our network predicts therefore the deformations of this subset of $l = 861$ vertices, which are then completed by our GP layer. The GP layer consists of 19 body part components that exploit Equation (6) with pre-computed kernel matrices. As explained before, the vertex of template is deformed by a point-wise decoder. While this is similar in spirit to [20] and [38], our approach differs in 2 aspects: (i) Instead of considering random points over the mesh surface during training, our approach focuses on a fixed subset of points – this allows us to better exploit the local spatial consistency of the reference mesh deformations; (ii) Instead of directly predicting the deformed template vertices, our point-wise decoder predicts the deformations (residuals) with respect to the template. The rationale here is that the residual space is generally easier to learn than the original coordinate space. In Figure 1, we show the segmented reference mesh and the 861 selected vertices deformed by the prediction of the point-wise decoder. Since the prediction is in the same order as the reference mesh, we can directly map the body part segmentation on the prediction of point-wise decoder.

Fully Connected Layer The previous GP layer enforces local spatial constraints between mesh vertices by assuming joint Gaussian distributions that can be pre-learned from a few meshes. In order to complete this with more global considerations over the vertices of a shape, we also employ a fully-connected multi-layer perceptron as another interpolation flow. This MLP takes as input the $l = 861$ deformed vertices as predicted by the point-wise decoder. It is composed of a hidden layer of dimension 2048, followed by 2 times hyperbolic tangent activation functions, and one linear layer to interpolate to the resolution of the reference mesh, in practice 6890 vertices with the SURREAL synthetic data.

3.3 Training Loss

In order to train our network we define a loss function $\mathcal{L}_r$ that accounts for the 3 outputs yielded by the decoder. The point-wise decoder computes the deformation field y_i over the subset of l mesh vertices on the reference mesh, while the GP and MLP layers output the deformed instances in the same resolution as the reference mesh. Hence:

$$\mathcal{L}_r(\hat{\mathrm{Y}}_{pw}, \mathrm{Y}_l, \hat{\mathrm{S}}_{gp}, \hat{\mathrm{S}}_{mlp}, \mathrm{S}) = L(\hat{\mathrm{Y}}_{pw}, \mathrm{Y}_l) + L(\hat{\mathrm{S}}_{gp}, \mathrm{S}) + L(\hat{\mathrm{S}}_{mlp}, \mathrm{S}) \tag{7}$$

where $L(\cdot, \cdot)$ denotes the standard mean-square error, $\hat{\mathrm{Y}}_{pw}$, $\hat{\mathrm{S}}_{gp}$, $\hat{\mathrm{S}}_{mlp}$ are the point-wise decoder, GP and MLP layer predictions respectively, Y_l is the ground truth deformation field over the reference mesh reduced to the l vertices predicted by the point-wise decoder and S is the ground truth deformed instance. In practice, we remark that the mesh obtained with the MLP layer is often blurry. However, the associated global constraint in the reconstruction loss appears to be beneficial in our experiments.

Adversarial Loss In addition to the loss presented in the previous section, we investigate in this work the contribution of introducing an adversarial strategy [33] in the proposed framework. While the previous loss function enforces local and more global spatial consistency, it does not encode knowledge on what a regular shape should be. Hence artifacts can occur when considering data outside the training set, as in Figure 4 with test data. In order to better detect abnormal outputs, we therefore propose an additional adversarial loss.

Recall that, given an arbitrary input point cloud P, the encoder generates a latent feature $\mathcal{Z}_p$. From this latent feature, the decoder generates a deformed version, $\hat{\mathrm{S}}$, of the reference mesh. In principle, feeding the encoder with this set $\hat{\mathrm{S}}$ should yield a latent feature $\mathcal{Z}_{\hat{\mathrm{s}}(p)}$ statistically similar to $\mathcal{Z}_p$. We therefore express the adversarial loss as:

$$\mathcal{L}_a(\mathrm{P}, \hat{\mathrm{S}}) = \mathbb{E}_p[log(D(\mathcal{Z}_p)] + \mathbb{E}_{\hat{\mathrm{s}}(p)}[log(1 - D(\mathcal{Z}_{\hat{\mathrm{s}}})] \tag{8}$$

where $D(\cdot)$ is the discriminator trained to detect abnormal latent features. It projects the 1024-dimension point feature into 512 and then 256 dimensions with two hidden layers, and outputs a probability. The two hidden layers are activated by an *ELU* (Exponential Linear Unit) function followed by batch normalization; the output is activated by a *sigmoid* non linearity. The final loss for our network training is a combination of $\mathcal{L}_r$ and $\mathcal{L}_a$:

$$\mathcal{L}_t = \lambda_1 \mathcal{L}_r + \lambda_2 \mathcal{L}_a. \tag{9}$$

The training algorithm proceeds by iteratively updating the encoder-decoder and the discriminator as depicted below. The protocol followed in practice is detailed in Section 4.

Algorithm 1: Training Algorithm

Input: Ground truth deformed instances S of the reference mesh
Initialization;
for *Training iterations* **do**
1. Sample a mini-batch of point cloud $\mathrm{P} \in \mathrm{S}$;
2. Compute the reconstruction $\hat{\mathrm{S}}(\mathrm{P})$;
3. Update $D(\cdot)$ by taking a learning step on loss $\mathcal{L}_a(\mathrm{P} \sim real, \hat{\mathrm{S}} \sim fake)$ (8);
4. Update then encoder and decoder by taking a learning step on loss $\mathcal{L}_t(\hat{\mathrm{S}} \sim real)$ (9);

end

4 Experimental Results

In this section, we first describe the datasets and the corresponding evaluation protocols. We then compare our approach against the state-of-the-art methods and provide a detailed analysis of our framework.

4.1 Datasets

We evaluate our framework for reconstructing human body meshes from point cloud data on the standard SURREAL [26] and FAUST [6] datasets.

The SURREAL dataset is a large-scale synthetic dataset that consists of textured human body shapes in different 3D poses. We follow the protocol introduced in [20] to generate our training data that consists of $230,000$ meshes.

The FAUST dataset provides 100 training and 200 testing human body scans of approximately 170000 vertices. They may include noise and holes, typically missing parts on the feet. The FAUST benchmark defines two challenges: the one on intra-, the other on inter-subject correspondences. We use the FAUST dataset only for testing purposes and do not use the provided scans for training.

4.2 Evaluation Protocol

We use the symmetric Chamfer distance between the predicted and ground-truth human shape to evaluate our framework on the SURREAL validation dataset. For our experiments on the FAUST dataset, we use the official test server to measure our accuracy. Throughout our experiments, we use the same training/test splits as 3D-CODED [20]. We perform a line-search to find the initial orientation and the initial translation that gives the smallest Chamfer distance during testing FAUST.

4.3 Implementation Details

We implement our Adversarial GP network in PyTorch and train for 25 epochs from scratch. In practice, we set $\lambda_1 = 10$ and $\lambda_2 = 0.05$. We use the Adam optimizer with a learning rate of 0.001 for the Discriminator and 0.0005 for Encoder and Decoder. We set the batch size to 32. We follow 3D-CODED [20] to add random translation between -3 cm and 3 cm to increase the robustness during training.

4.4 Comparison with Baselines

We report reconstruction and registration accuracy on the SURREAL [26] and FAUST [6] datasets and compare our results to the state-of-the-art results of [20] and [38] in Table 1.

We further use the following baselines and versions of our approach in the evaluation:

- *MLP*: A multi-layer perceptron with 2 layers as described in Section 3.2 operating on the output deformations of the point-wise decoder.
- *GP*: Gaussian Process layer as described in Section 3.2 operating on the output deformations of the point-wise decoder.
- *Adversarial GP*: Adversarial network coupled with the Gaussian process and MLP layers that operates on the output deformations of the point-wise decoder (see Section 3.3).

Method	SURREAL-Chamfer	FAUST-inter
3D-CODED[20]	1.33	2.88
AtlasNet2-Deformation[38] 3D	1.17	2.76
AtlasNet2-Points 3D	1.11	3.05
AtlasNet2-Deformation 10D	1.01	2.77
AtlasNet2-Points 10D	1.01	2.85
Ours(MLP)	0.54	2.94
Ours(GP)	**0.35**	**2.73**
Ours(Adversarial GP)	0.50	2.76

Table 1. Results on the SURREAL validation set for human body reconstruction and on the FAUST-inter correspondence challenge. As in [38, 20], we report the symmetric Chamfer distance ($\times 10^{-3}$) for SURREAL validation. For FAUST, we report the Euclidean correspondence error in (cm). In FAUST, we apply the same refinement technique as in 3D-CODED to our MLP, GP and Adversarial GP.

We further compare our results to [38, 20] qualitatively to demonstrate the effectiveness of our method in Figure 3 and Figure 4.

Reconstruction. We report our surface reconstruction results in comparison to [20, 38] on the SURREAL and FAUST datasets in Table 1. While providing accurate reconstructions, [20] relies on point-to-point distance minimization, therefore lacking global context. To remedy this and encode global context, we apply an MLP on point-wise predictions. This would help encode global context, but in return, would ignore local dependencies. Our GP layer, on the other hand, aims at finding a local context on each body part. As can be seen in Table 1, the GP layer yields the most accurate reconstruction results on the SURREAL validation set and in the FAUST Inter-Subject challenge. In Figure 3, we also show qualitative results on SURREAL validation of the variants of our approach (MLP, GP, Adversarial GP) in comparison to 3D-CODED [20] and AtlasNet2 [38]. Our method yields better reconstruction accuracy than [20] and [38] and provides realistic surface reconstructions

Registration. Our output mesh is reconstructed from an input point cloud and is aligned with a template shape. Therefore, our method could further compute registration to the human body by finding the closest point on the reconstruction. We evaluate our method on the FAUST [6] challenge, that includes 100-pairs of shapes to be matched. In FAUST, the input is real scan data in different orientations and translations and scans typically include noise and holes. In Table 2, we report the results of all published studies to date on the FAUST challenge. We do not include the results of DHNN as it requires manual selection of additional landmark points which is used to guide the optimization.

Method	Intra (cm)	Inter (cm)
3D-CODED[20]	1.985	2.878
Stitched puppets[8]	1.568	3.126
LBS-AE[16]	2.161	4.08
FARM[18]	2.810	4.123
BPS[19]	2.327	4.529
FMNet[13]	2.436	4.826
Convex-Opt[41]	4.860	8.304
Our GP	2.349	2.734
Our Adversarial GP	1.904	2.759

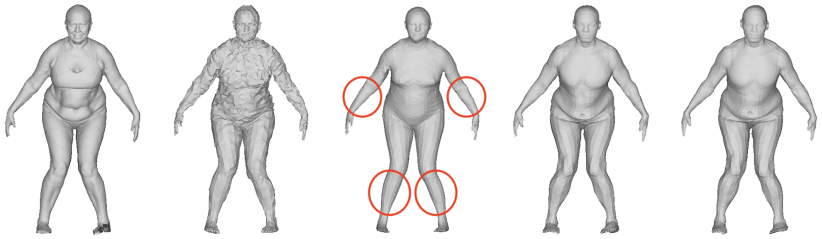
Table 2. Results for the FAUST intra- and inter-subject challenges for human body registration.

Importance of Adversarial Training. Although our GP network provides accurate reconstruction and registration results, we have observed in practice that it sometimes results in artifacts, as can be seen in a few cases in Figure 4. Our adversarial GP, on the other hand, is able to correct these artifacts and results in physically plausible human shape reconstructions, as demonstrated in Figure 4. This is in part due to the fact that adversarial training prevents overfitting to the SURREAL training data and achieves good generalization across datasets. We have observed that using the MLP network along with the GP layer further regularizes the training of our Adversarial GP framework. Therefore, in practice, we also employ an MLP during training of our Adversarial GP.

Method	SURREAL-Chamfer	FAUST-intra	FAUST-inter
Adv+GP (w.o. MLP)	0.52	2.585	2.913
MLP+GP (w.o. Adv)	**0.37**	2.042	2.858
MLP+GP+L2 weight decay	5.40	6.068	7.58
MLP+GP+Dropout	0.38	2.236	2.984
Adv+MLP+GP (Adv GP)	0.50	**1.904**	**2.759**

Table 3. Numeric comparisons. We report the symmetric Chamfer distance ($\times 10^{-3}$) on the SURREAL validation dataset and Euclidean correspondence error (cm) in FAUST -intra/-inter challenges for the variants of our model. We further compare adversarial training to L2 weight decay (regularization term $\lambda = 5 \times 10^{-4}$) and dropout. See more qualitative results in the supplementary material.

In Table 3, we further analyze the influence of adversarial loss on the reconstruction and registration. Using an adversarial loss yields more accurate results on the FAUST dataset. While resulting in lower accuracy on the SURREAL dataset, adversarial training helps to prevent overfitting by ensuring that the distributions of the input data and reconstruction are similar. In Figure 4, we



(a) input scan (b) MLP (c) MLP+Laplace (d) GP (e) Adv GP

Fig. 2. Smoothness of GP. From left to right, (a) input scan, reconstruction in standard resolution of (b) MLP, (c) MLP smoothened by the Laplacian operator, (d) GP, and (e) Adversarial GP.

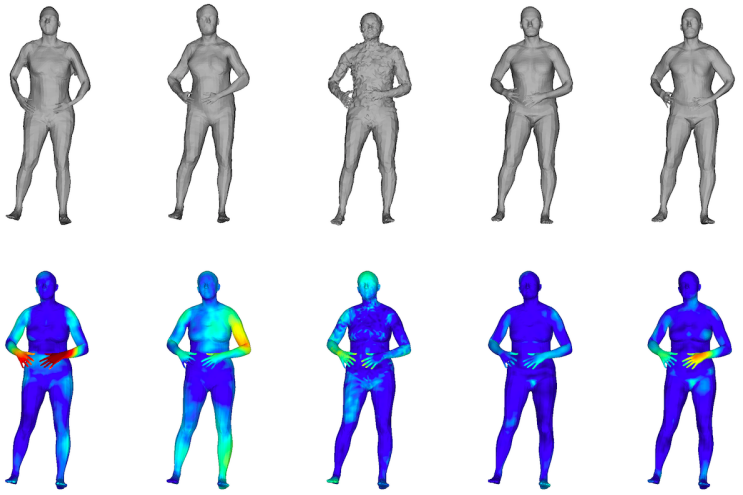
demonstrate that adversarial training in practice results in physically more plausible and realistic shapes. To demonstrate the effectiveness of adversarial training as a regularization mechanism, we further compare it to standard regularization techniques of L2-weight decay and dropout in Table 3.

Influence of Gaussian Kernel Regularization. In Figure 2, we present qualitative reconstruction results obtained with different decoders to further support our quantitative analysis in Table 1. While the MLP decoder results in a blurry shape, Laplacian denoising results in a shrinkage in the volume, especially in the limbs. GP and Adversarial GP, on the other hand, provide high-fidelity reconstructions.

Method	3D-CODED	AtlasNet2	GP	Adversarial GP
without refinement	6.29	4.72	**4.71**	4.964
with refinement	3.048	-	**2.734**	2.873
with refinement+ high-res template	2.878	2.76	2.815	**2.759**

Table 4. Comparison to 3D-CODED [20] and AtlasNet2 [38] with and without refinement. We report Euclidean correspondence errors on the FAUST-inter challenge in (cm). The refinement is based on optimizing the global feature to minimize the symmetric Chamfer distance. We follow [20] to register the scan to a high-resolution template.

Refinement. During evaluation, we follow the same refinement strategy of 3D-CODED [20], that minimizes the Chamfer distance between reconstructions and inputs. Consequently, a nearest neighbor search is performed to find correspondences and match shapes. To highlight the benefit of refinement, we show in



(a) [20] 3.86/15.15 (b) [38] 3.03/7.99 (c) MLP 2.73/7.11 (d) GP 1.93/5.15 (e) Adv GP 2.36/7.76

Fig. 3. Qualitative evaluation for human shape reconstruction. From left to right, reconstruction in standard resolution of (a) 3D-CODED [20], (b) AtlasNet2 [38], (c) Our MLP, (d) our GP and (e) our Adversarial GP. And we report the heatmaps and mean/max Euclidean reconstruction error in (cm) for this instance.

Table 4 our results in comparison to [20] and [38] with and without refinement. Refinement results in better accuracy for our method, as expected, and our approach provides better results in comparison to [20] and [38] in all cases. When we use a high resolution template for the nearest neighbor step, we gain an additional accuracy improvement for Adversarial GP, but not for GP. The result could not be always improved by using a high resolution template due to the fact that the FAUST-inter challenge computes the Euclidean distance between the prediction and sparse landmarks. Since the Euclidean distance is more tolerant of the artifacts in Figure 4 than geodesic distance, Adversarial GP can not make great improvement in FAUST challenge.

5 Conclusion

We have presented an encoder-decoder neural network architecture to reconstruct human body meshes from point cloud data, by learning dense human body correspondences. Our architecture enforces surface consistency with a specialized Gaussian process layer. Our adversarial training framework allows for generalization across datasets and reconstructs high-fidelity human meshes. Future work will apply the proposed framework to problems like motion sequence alignment and tracking.

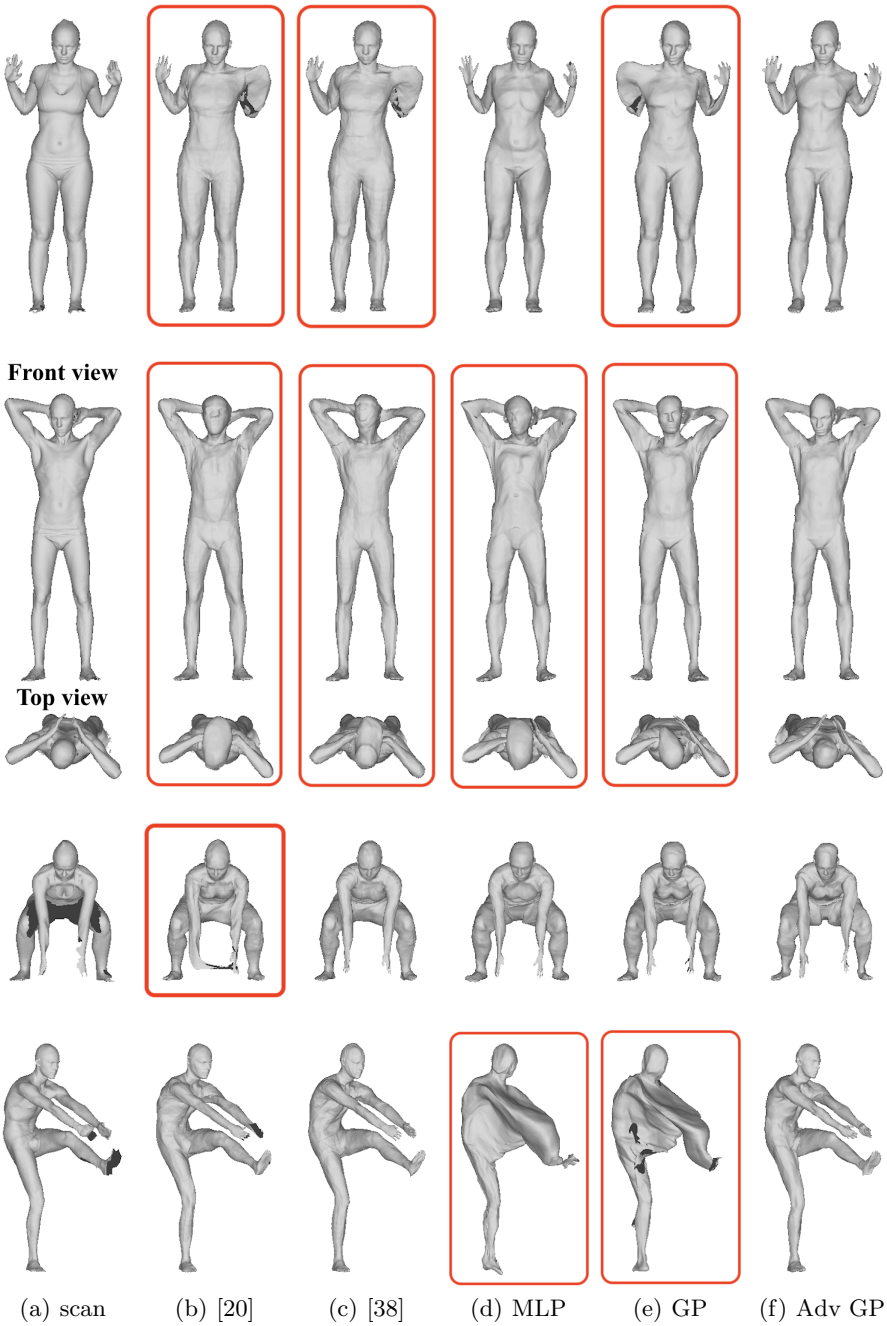


Fig. 4. Challenging cases in FAUST. From left to right, (a) input scan, reconstruction in high resolution of (b) 3D-CODED [20], (c) AtlasNet2 [38], (d) Our MLP, (e) our GP and (f) our Adversarial GP. We highlight the failure cases with red box.

References

1. Bronstein, A.M., Bronstein, M.M., Kimmel, R.: Generalized multidimensional scaling: A framework for isometry-invariant partial surface matching. Proceedings of the National Academy of Sciences **103** (2006) 1168–1172
2. Ovsjanikov, M., Mérigot, Q., Mémoli, F., Guibas, L.: One point isometric matching with the heat kernel. Computer Graphics Forum **29** (2010) 1555–1564
3. Kim, V., Lipman, Y., Funkhouser, T.: Blended intrinsic maps. ACM Transactions on Graphics (TOG) **30** (2011)
4. Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., Davis, J.: Scape: shape completion and animation of people. ACM Transactions on Graphics (TOG) **24** (2005) 408–416
5. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: SMPL: A skinned multi-person linear model. ACM Transaction on Graphics (TOG) **34** (2015) 248:1–248:16
6. Bogo, F., Romero, J., Loper, M., Black, M.J.: Faust: Dataset and evaluation for 3d mesh registration. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2014) 3794–3801
7. Bogo, F., Romero, J., Pons-Moll, G., Black, M.J.: Dynamic faust: Registering human bodies in motion. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 6233–6242
8. Zuffi, S., Black, M.J.: The stitched puppet: A graphical model of 3d human shape and pose. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 3537–3546
9. Rodolà, E., Bulò, S., Windheuser, T., Vestner, M., Cremers, D.: Dense non-rigid shape correspondence using random forests. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2014) 4177–4184
10. Monti, F., Boscaini, D., Masci, J., Rodola, E., Svoboda, J., Bronstein, M.M.: Geometric deep learning on graphs and manifolds using mixture model CNNs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 5115–5124
11. Wei, L., Huang, Q., Ceylan, D., Vouga, E., Li, H.: Dense human body correspondences using convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 1544–1553
12. Halimi, O., Litany, O., Rodola, E., Bronstein, A.M., Kimmel, R.: Unsupervised learning of dense shape correspondence. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2019) 4370–4379
13. Litany, O., Remez, T., Rodola, E., Bronstein, A., Bronstein, M.: Deep functional maps: Structured prediction for dense shape correspondence. In: Proceedings of the IEEE International Conference on Computer Vision. (2017) 5659–5667
14. Tombari, F., Salti, S., Di Stefano, L.: Unique signatures of histograms for local surface description. In: Proceedings of the European Conference on Computer Vision, Springer (2010) 356–369
15. Tan, Q., Gao, L., Lai, Y.K., Xia, S.: Variational autoencoders for deforming 3d mesh models. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 5841–5850
16. Li, C.L., Simon, T., Saragih, J., Póczos, B., Sheikh, Y.: Lbs autoencoder: Self-supervised fitting of articulated meshes to point clouds. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2019) 11967–11976

17. Jiang, B., Zhang, J., Cai, J., Zheng, J.: Disentangled human body embedding based on deep hierarchical neural network. In: IEEE Transactions on Visualization and Computer Graphics. Volume 26. (2020) 2560–2575
18. Marin, R., Melzi, S., Rodolà, E., Castellani, U.: FARM: Functional automatic registration method for 3d human bodies. In: Computer Graphics Forum. Volume 39., Wiley Online Library (2020) 160–173
19. Prokudin, S., Lassner, C., Romero, J.: Efficient learning on point clouds with basis point sets. In: Proceedings of the IEEE International Conference on Computer Vision. (2019) 4332–4341
20. Groueix, T., Fisher, M., Kim, V.G., Russell, B., Aubry, M.: 3d-coded : 3d correspondences by deep deformation. In: Proceedings of the European Conference on Computer Vision. (2018) 235–251
21. Groueix, T., Fisher, M., Kim, V.G., Russell, B., Aubry, M.: AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 216–224
22. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 652–660
23. Hu, Y., Gibson, E., Ghavami, N., Bonmati, E., Moore, C.M., Emberton, M., Vercauteren, T., Noble, J.A., Barratt, D.C.: Adversarial deformation regularization for training image registration neural networks. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer (2018) 774–782
24. Biasotti, S., Cerri, A., Bronstein, A., Bronstein, M.: Recent trends, applications, and perspectives in 3d shape similarity assessment. Computer Graphics Forum **36** (2016) 87–119
25. van Kaick, O., Hamarneh, G., Cohen-Or, D.: A survey on shape correspondence. Computer Graphics Forum **30** (2011) 1681–1707
26. Varol, G., Romero, J., Martin, X., Mahmood, N., Black, M.J., Laptev, I., Schmid, C.: Learning from synthetic humans. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 4627–4635
27. Magnenat-Thalmann, N., Laperrire, R., Thalmann, D.: Joint-dependent local deformations for hand animation and object grasping. In: Proceedings of Graphics Interface. Volume '88., Citeseer (1988)
28. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 4700–4708
29. Lüthi, M., Gerig, T., Jud, C., Vetter, T.: Gaussian process morphable models. IEEE transactions on pattern analysis and machine intelligence **40** (2017) 1860–1873
30. Wilson, A.G., Knowles, D.A., Ghahramani, Z.: Gaussian process regression networks. In: Proceedings of the International Conference on Machine Learning. (2011)
31. Wilson, A.G., Hu, Z., Salakhutdinov, R., Xing, E.P.: Deep kernel learning. In: Artificial Intelligence and Statistics. (2016) 370–378
32. Damianou, A., Lawrence, N.: Deep gaussian processes. In: Artificial Intelligence and Statistics. (2013) 207–215
33. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems. (2014) 2672–2680

34. Kanazawa, A., Black, M.J., Jacobs, D.W., Malik, J.: End-to-end recovery of human shape and pose. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 7122–7131
35. Ma, Q., Yang, J., Ranjan, A., Pujades, S., Pons-Moll, G., Tang, S., Black, M.J.: Learning to dress 3d people in generative clothing. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2020)
36. Fernández Abrevaya, V., Boukhayma, A., Wuhrer, S., Boyer, E.: A decoupled 3d facial shape model by adversarial training. In: Proceedings of the IEEE International Conference on Computer Vision. (2019) 1–10
37. Shamai, G., Slossberg, R., Kimmel, R.: Synthesizing facial photometries and corresponding geometries using generative adversarial networks. ACM Transactions on Multimedia Computing, Communications, and Applications **15** (2019) 1–24
38. Deprelle, T., Groueix, T., Fisher, M., Kim, V., Russell, B., Aubry, M.: Learning elementary structures for 3d shape generation and matching. In: Advances in Neural Information Processing Systems. (2019) 7435–7445
39. Williams, C.K., Rasmussen, C.E.: Gaussian processes for machine learning. MIT press Cambridge, MA (2006)
40. Basset, J., Wuhrer, S., Boyer, E., Multon, F.: Contact Preserving Shape Transfer For Rigging-Free Motion Retargeting. In: MIG 2019 - ACM SIGGRAPH Conference Motion Interaction and Games. (2019) 1–10
41. Chen, Q., Koltun, V.: Robust nonrigid registration by convex optimization. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 2039–2047