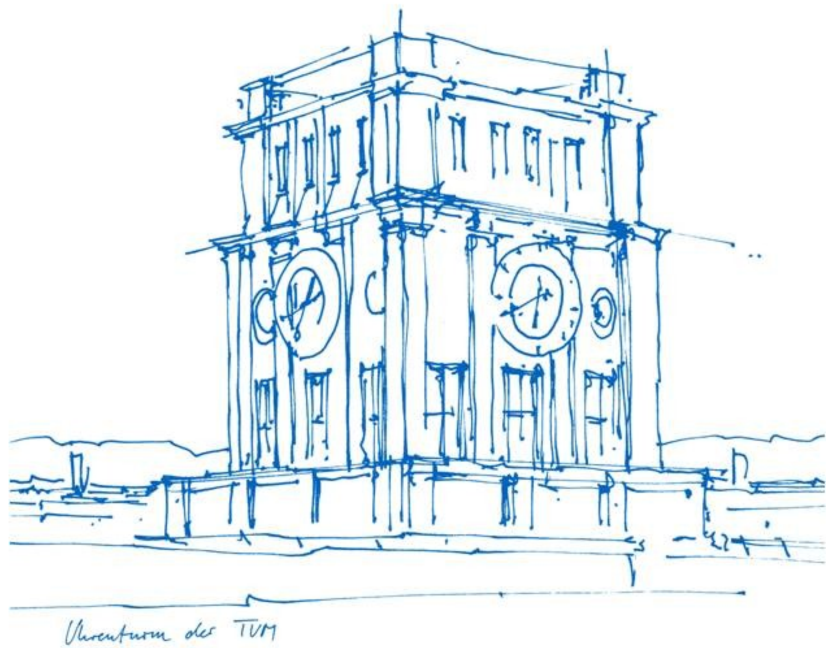
Learning the Human Distribution

Master Seminar Robot Perception & Intelligence

Xiaoyue Hu

16. January 2024



Outline

- Introduction
- Related works
- Method descriptions
- Experiments and results
- Future work

Introduction

- Human pose prior
- What is a human pose prior?
- Why do we need human pose prior?

SMPL(A Skinned Multi-Person Linear Model)

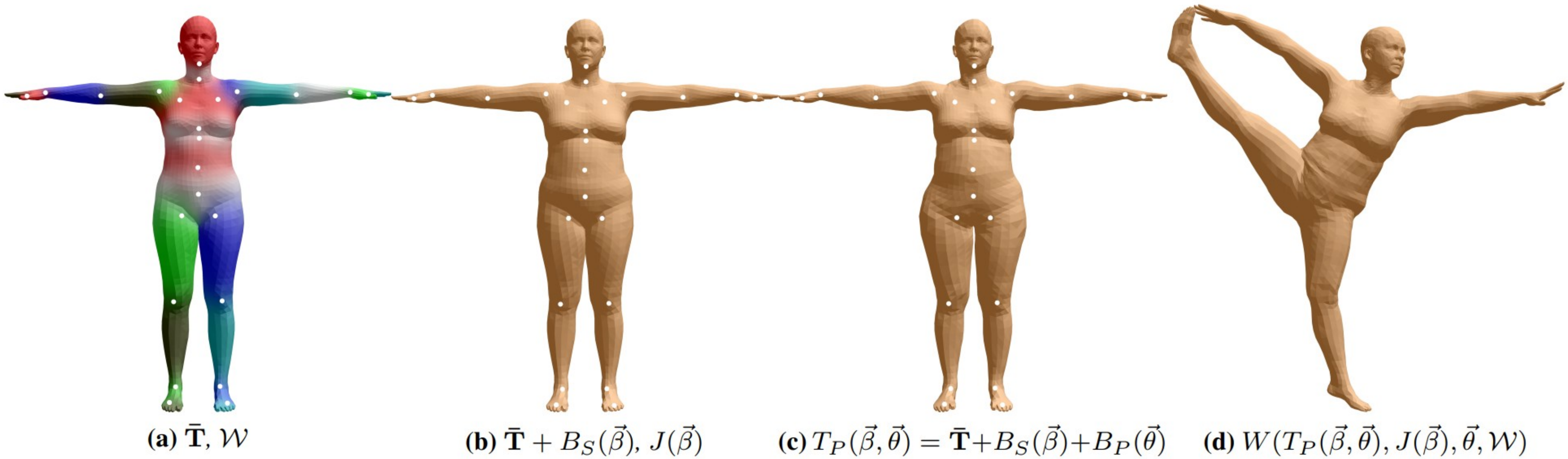


Figure 1: SMPL model[1]

3D mesh with N = 6890 vertices and K = 23(24) joints

$$
\begin{aligned}
M(\vec{\beta}, \vec{\theta}) &= W(T_P(\vec{\beta}, \vec{\theta}), J(\vec{\beta}), \vec{\theta}, \mathcal{W}) \\
T_P(\vec{\beta}, \vec{\theta}) &= \bar{\mathbf{T}} + B_S(\vec{\beta}) + B_P(\vec{\theta})
\end{aligned}
$$

$B_S(\vec{\beta})$: a blend shape function

$J(\vec{\beta})$: a function to predict K joint locations

$B_P(\vec{\theta})$: a pose-dependent blend shape function

[1] SMPL: A Skinned Multi-Person Linear Model

SMPLify

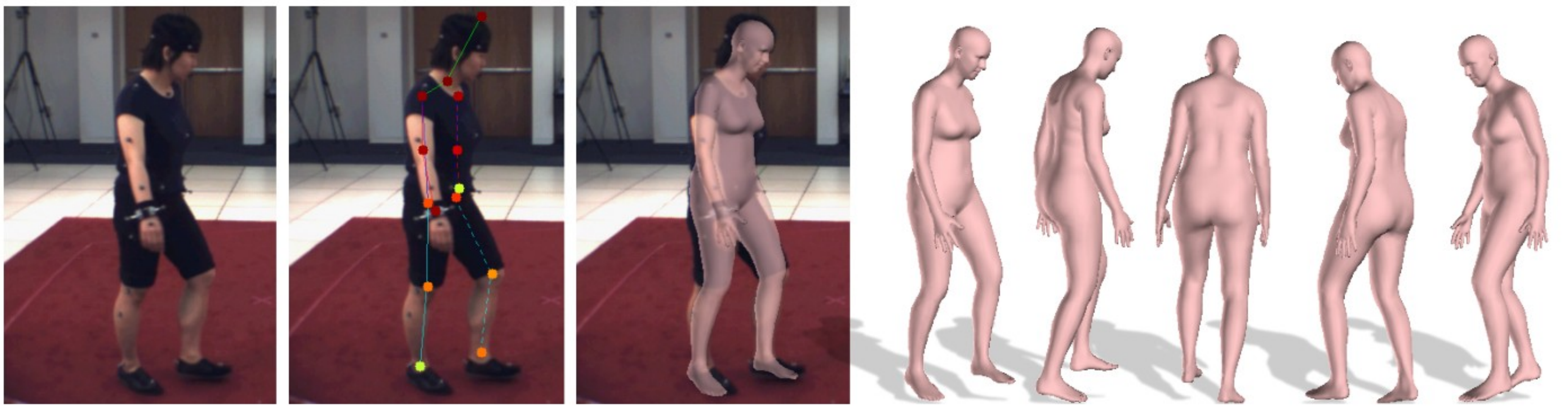


Fig. 2. SMPLify System overview[2]

- CNN-based method to predict 2D joint locations(DeepCut)
- fit a 3D body model to predicted 2D joints

[2] Keep it SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image

SMPLify

Loss function

$$E(\boldsymbol{\beta}, \boldsymbol{\theta}) = E_J(\boldsymbol{\beta}, \boldsymbol{\theta}; K, J_{\text{est}}) + \lambda_\theta E_\theta(\boldsymbol{\theta}) + \lambda_a E_a(\boldsymbol{\theta}) + \lambda_{sp} E_{sp}(\boldsymbol{\theta}; \boldsymbol{\beta}) + \lambda_\beta E_\beta(\boldsymbol{\beta})$$

$$E_J(\boldsymbol{\beta}, \boldsymbol{\theta}; K, J_{\text{est}}) = \sum_{\text{joint } i} w_i \rho(\Pi_K(R_\theta(J(\boldsymbol{\beta})_i)) - J_{\text{est},i})$$

$$\begin{aligned} E_\theta(\boldsymbol{\theta}) \equiv -\log \sum_j (g_j \mathcal{N}(\boldsymbol{\theta}; \boldsymbol{\mu}_{\theta,j}, \Sigma_{\theta,j})) &\approx -\log(\max_j(c g_j \mathcal{N}(\boldsymbol{\theta}; \boldsymbol{\mu}_{\theta,j}, \Sigma_{\theta,j}))) \\ &= \min_j \left(-\log(c g_j \mathcal{N}(\boldsymbol{\theta}; \boldsymbol{\mu}_{\theta,j}, \Sigma_{\theta,j}))\right) \end{aligned}$$

$$E_a(\boldsymbol{\theta}) = \sum_i \exp(\boldsymbol{\theta}_i)$$

[2] Keep it SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image

SMPLify

Fig. 3. Example results[2]

[2] Keep it SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image

Human pose prior(VPoser)

Variational autoencoder(VAE based)

Extend SMPL to SMPL-X

The new SMPL-X model has N = 10475 vertices and K = 54 joints

Extend SMPLify to SMPLify-X

$$E(\beta,\theta,\psi) = E_J + \lambda_{\theta_b} E_{\theta_b} + \lambda_{\theta_f} E_{\theta_f} + \lambda_{m_h} E_{m_h} + \lambda_\alpha E_\alpha + \lambda_\beta E_\beta + \lambda_{\mathcal{E}} E_{\mathcal{E}} + \lambda_{\mathcal{C}} E_{\mathcal{C}}$$

$$E_J(\boldsymbol{\beta},\boldsymbol{\theta};K,J_{\text{est}}) = \sum_{\text{joint } i} w_i \rho(\Pi_K(R_\theta(J(\boldsymbol{\beta})_i)) - J_{\text{est},i})$$

$$E_\alpha(\theta_b) = \sum_{i\in(elbows,knees)} \exp(\theta_i)$$

What about $E_{\theta_b}(\theta_b)$

[3] Expressive Body Capture: 3D Hands, Face, and Body from a Single Image

Human pose prior(VPoser)

Variational autoencoder(VAE based)

$$\mathcal{L}_{total} = c_1\mathcal{L}_{KL} + c_2\mathcal{L}_{rec} + c_3\mathcal{L}_{orth} + c_4\mathcal{L}_{det1} + c_5\mathcal{L}_{reg}$$

$$\mathcal{L}_{KL} = KL(q(Z|R)||\mathcal{N}(0, I))$$
$$\mathcal{L}_{rec} = ||R - \hat{R}||_2^2$$
$$\mathcal{L}_{orth} = ||\hat{R}\hat{R}^{'} - I||_2^2$$
$$\mathcal{L}_{det1} = |det(\hat{R}) - 1|$$
$$\mathcal{L}_{reg} = ||\phi||_2^2,$$

Where $Z \in \mathbb{R}^{32}$ $R \in SO(3)$ $E_{\theta_b}(\theta_b) = ||z||_2^2$

[3] Expressive Body Capture: 3D Hands, Face, and Body from a Single Image

VPoser Experiments

Table 1

Model	Keypoints	v2v error	Joint error
“SMPL”	Body	57.6	63.5
“SMPL”	Body+Hands+Face	64.5	71.7
“SMPL+H”	Body+Hands	54.2	63.9
SMPL-X	Body+Hands+Face	52.9	62.6

both fit on EHF dataset
(Expressive Hands and Face)

Table 2

Version	v2v error
SMPLify-X	52.9
gender neutral model	58.0
replace Vposer with GMM	56.4
no collision term	53.5

[3] Expressive Body Capture: 3D Hands, Face, and Body from a Single Image

VPoser Experiments

Fig. 4. Example results[3]

[3] Expressive Body Capture: 3D Hands, Face, and Body from a Single Image

VPoser Experiments

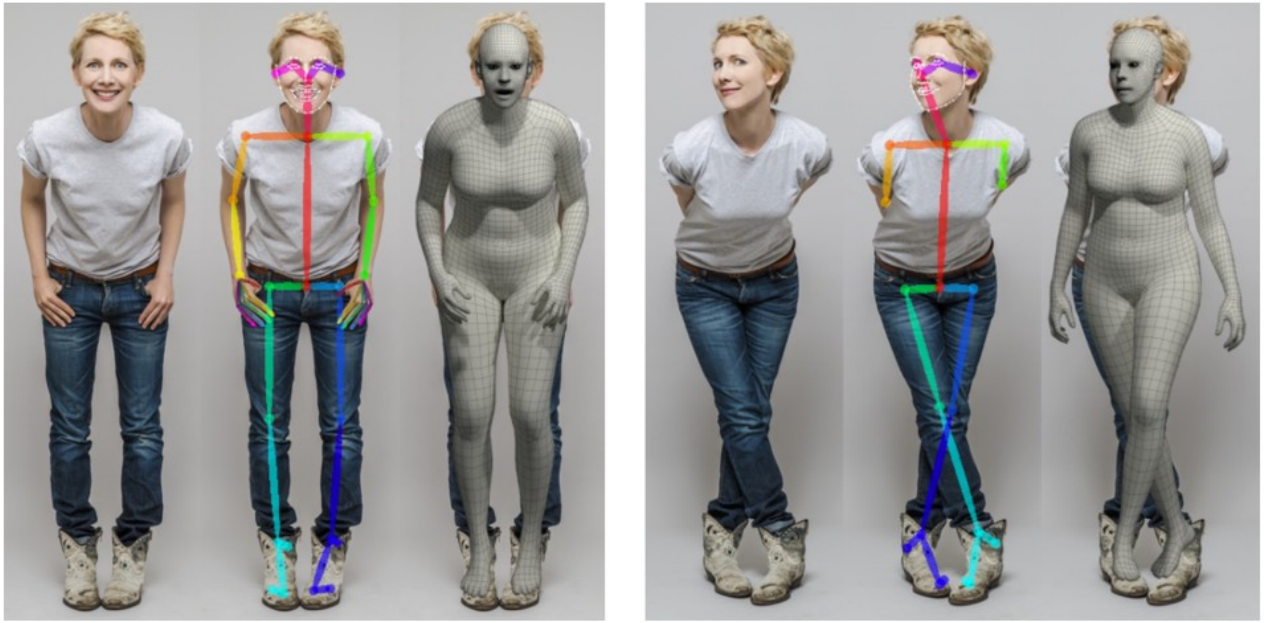


Fig. 5. Failure cases[3]

Human pose prior(Pose-NDF)

- High-dimensional domain $SO(3)^K$
- More robust
- Fully differentiable
- More diverse samples

3D representation based on SMPL

Human pose prior(Pose-NDF)

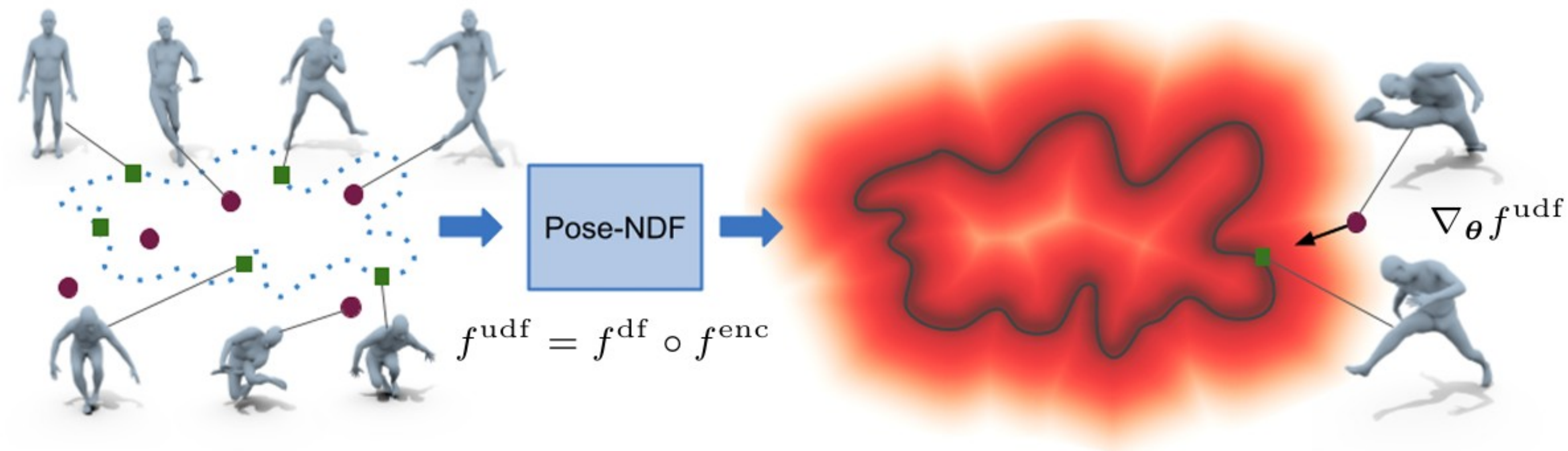


Fig. 6. Pose-NDF[4]

$$\mathcal{S} = \{\boldsymbol{\theta} \in SO(3)^K \mid f(\boldsymbol{\theta}) = 0\}$$

[4] Pose-NDF: Modeling Human Pose Manifolds with Neural Distance Fields

Human pose prior(Pose-NDF)

Distance between two poses:

$$d(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}) = \sqrt{\sum_{i=1}^{K} \frac{w_i}{2} (\arccos |\boldsymbol{\theta}_i^\top \cdot \hat{\boldsymbol{\theta}}_i|)^2},$$

$$\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_K\} \qquad \hat{\boldsymbol{\theta}} = \{\hat{\boldsymbol{\theta}}_1, ..., \hat{\boldsymbol{\theta}}_K\}$$

Hierarchical implicit function:

$$f_1^{\text{enc}} : (\boldsymbol{\theta}_1) \mapsto \mathbf{v}_1 \qquad f_k^{\text{enc}} : (\boldsymbol{\theta}_k, \mathbf{v}_{\tau(k)}) \mapsto \mathbf{v}_k, \quad k \in \{2 \dots K\}$$

$$\mathbf{p} = [\mathbf{v}_1 || \dots || \mathbf{v}_K]$$

$$f^{\text{df}} : \mathbb{R}^{l \cdot K} \to \mathbb{R}^+$$

$$f^{\text{udf}}(\boldsymbol{\theta}) = (f^{\text{df}} \circ f^{\text{enc}})(\boldsymbol{\theta})$$

[4] Pose-NDF: Modeling Human Pose Manifolds with Neural Distance Fields

Human pose prior(Pose-NDF)

Training the implicit function

Loss function:

$$\mathcal{L}_{\mathrm{UDF}} = \sum_{(\boldsymbol{\theta},d)\in\mathcal{D}} ||f^{\mathrm{udf}}(\boldsymbol{\theta}) - d_{\boldsymbol{\theta}}||_2 \quad \mathcal{L}_{\mathrm{eikonal}} = \sum_{(\boldsymbol{\theta},d)\in\mathcal{D},\ d\neq 0} (||\nabla_{\boldsymbol{\theta}} f^{\mathrm{udf}}(\boldsymbol{\theta})|| - 1)^2$$

$$\mathcal{D} = \{(\boldsymbol{\theta}_i, d_i)\}_{1\leq i\leq N}$$

Projection Algorithm:

$$\boldsymbol{\theta}^i = \boldsymbol{\theta}^{i-1} - \alpha f(\boldsymbol{\theta}^{i-1}) \nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}^{i-1})$$

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}\in SO(3)^K}{\arg\min}\ d(\boldsymbol{\theta}, \mathcal{S})$$

[4] Pose-NDF: Modeling Human Pose Manifolds with Neural Distance Fields

Pose-NDF Experiments

Motion denoising

Table 3

Data	HPS [23]			AMASS [38]			Noisy AMASS		
# frames	60	120	240	60	120	240	60	120	240
Method									
VPoser [49]	4.91	4.16	3.81	1.52	1.55	1.47	8.96	9.13	9.15
HuMoR [52]	9.69	8.73	10.86	3.21	3.62	3.67	11.04	17.14	30.31
Pose-NDF	**2.32**	**2.14**	**2.11**	**0.59**	**0.55**	**0.54**	**7.96**	**8.31**	**8.46**

Pose-NDF Experiments

Motion denoising

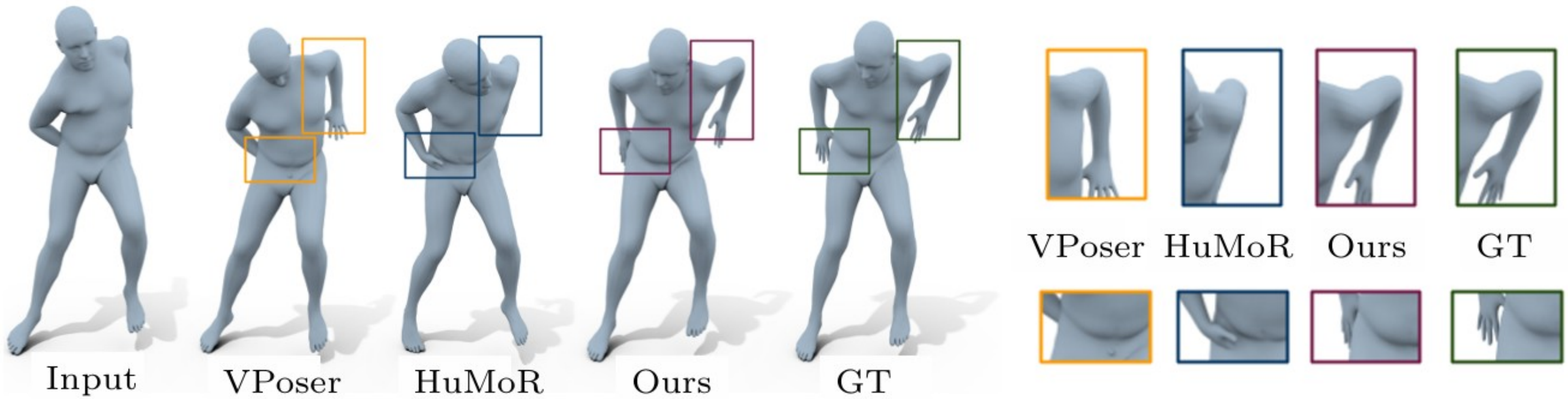


Fig. 7. motion denoising results[4]

[4] Pose-NDF: Modeling Human Pose Manifolds with Neural Distance Fields

Pose-NDF Experiments

Occlusion

Table 4

Data	Occ. Leg			Occ. Arm+hand			Occ. Shoulder +Upper Arm		
# frames	60	120	240	60	120	240	60	120	240
Method									
VPoser [49]	2.53	2.57	2.54	8.51	8.52	8.59	9.98	9.49	9.48
HuMoR [52]	5.60	6.19	9.09	7.83	8.44	10.25	**4.75**	**5.11**	**4.95**
Pose-NDF	**2.49**	**2.51**	**2.47**	**7.81**	**8.13**	**7.98**	7.63	7.89	6.76

[4] Pose-NDF: Modeling Human Pose Manifolds with Neural Distance Fields

Pose-NDF Experiments

3D pose estimation from images

Fig. 8. 3D shape and pose estimation[4]

[4] Pose-NDF: Modeling Human Pose Manifolds with Neural Distance Fields

Pose-NDF Experiments

3D pose estimation from images

$$\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\beta},\boldsymbol{\theta}}{\arg\min}\, \mathcal{L}_J + \lambda_{\boldsymbol{\theta}}\mathcal{L}_{\boldsymbol{\theta}} + \lambda_{\boldsymbol{\beta}}\mathcal{L}_{\boldsymbol{\beta}} + \lambda_{\alpha}\mathcal{L}_{\alpha}$$

$$\mathcal{L}_{\theta} \; = \; f^{\mathrm{udf}}(\boldsymbol{\theta})$$

$$\lambda_{\boldsymbol{\theta}} = w f^{\mathrm{udf}}(\boldsymbol{\theta})$$

Pose-NDF Experiments

3D pose estimation from images

Table 5

Method	Optimization			ExPose	ExPose + Optimization			
	VPoser [49]	GAN-S [16]	Pose-NDF	-	+No prior	+ VPoser [49]	+ GAN-S [16]	+**Pose-NDF**
Per-vertex error (mm)	60.34	59.18	57.39	54.76	99.78	67.23	54.09	53.81

[4] Pose-NDF: Modeling Human Pose Manifolds with Neural Distance Fields

Pose-NDF Experiments

3D pose generation

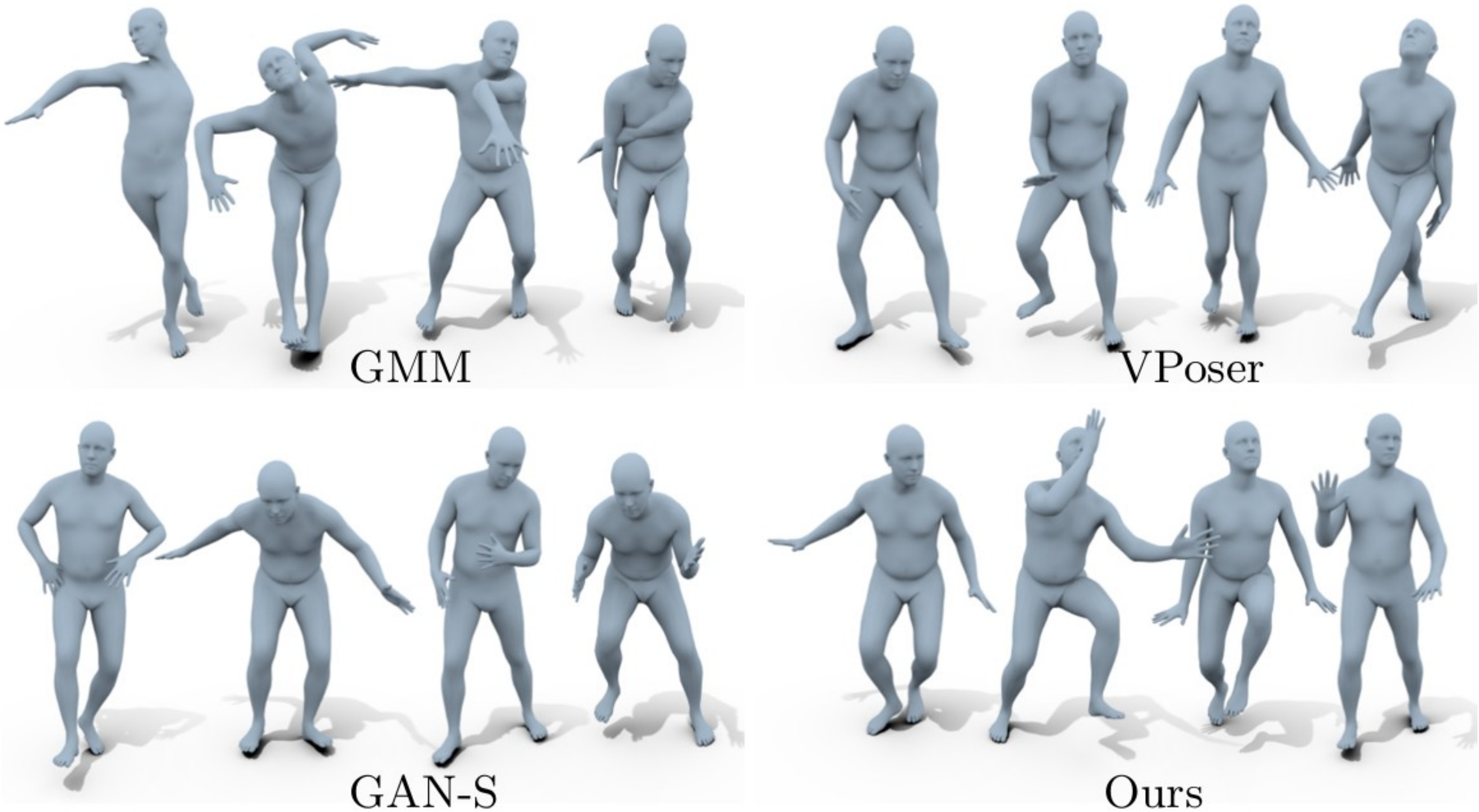


Fig. 9. pose generation[4]

[4] Pose-NDF: Modeling Human Pose Manifolds with Neural Distance Fields

Future work

- Rerson-ground Contacts(HuMoR)
- Learning parameters directly from images(extended model)
- Dynamic Dataset(motion)
- Moving cameras

HuMoR

Related papers

[1] SMPL: A Skinned Multi-Person Linear Model

[2] Keep it SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image

[3] Expressive Body Capture: 3D Hands, Face, and Body from a Single Image

[4] Pose-NDF: Modeling Human Pose Manifolds with Neural Distance Fields

[5] HuMoR: 3D Human Motion Model for Robust Pose Estimation

Thank you for listening