北京大學
PEKING UNIVERSITY

Pose-NDF: Modeling Human Pose Manifolds with Neural Distance Fields

Garvita Tiwari, Dimitrije Anti´c, Jan Eric Lenssen, Nikolaos Sarafianos,
Tony Tung3, and Gerard Pons-Moll

ECCV 2022 Oral - Best Paper Honourable Mention

PRESENTER: YUERU JIA

2023/02/19

Outline

1 Authors

2 Background

3 Method

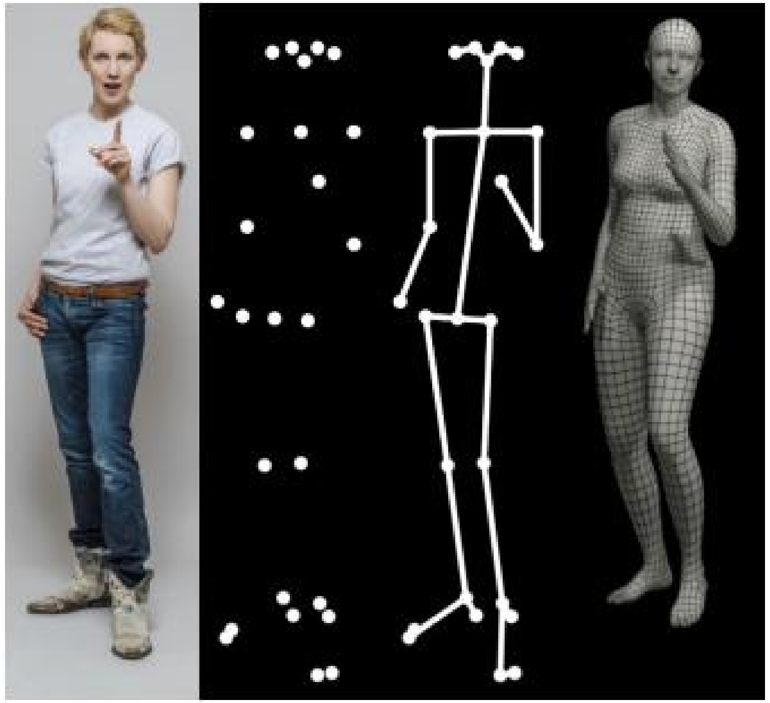
4 Experiments

5 Conclusion

Outline

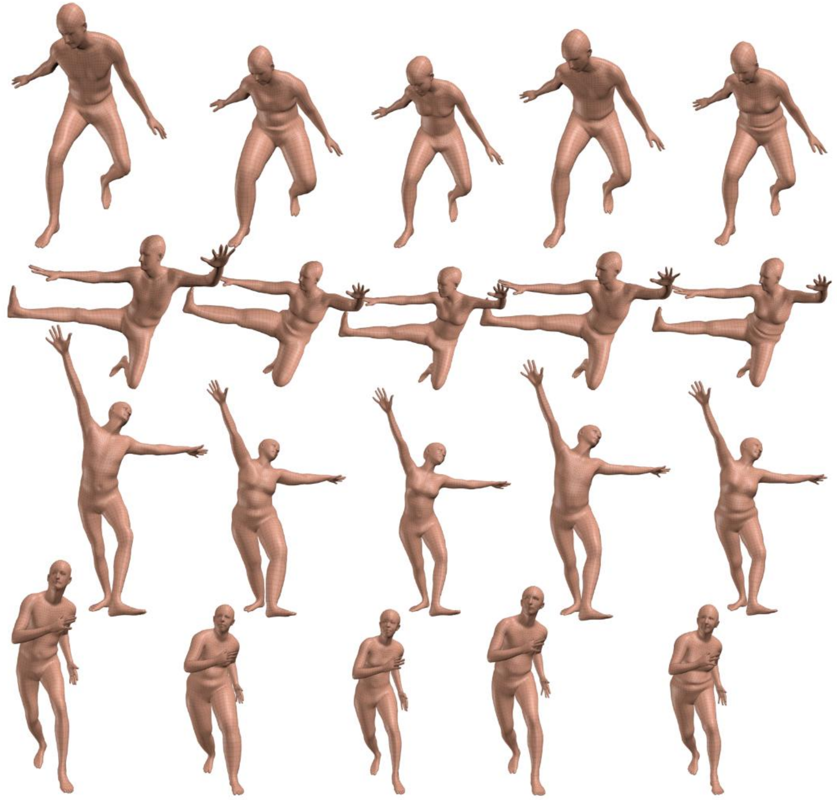
Background

- What is this work for?
- Pose and Motion Priors
  - estimate human pose from images and videos
  - data generation
  - denoising



Background

- SMPL (Skinned Multi-Person Linear Model)
  - vertex-based
  - shape: β , pose: θ
  - K=23, N=6890



Background

- Current methods
  - VAE-based: VPoser, HuMoR
  - Gaussian assumption’s limitations:
    - producing more likely poses near the mean of the computed Gaussian
    - Distances are not preserved
    - Dead regions

Background

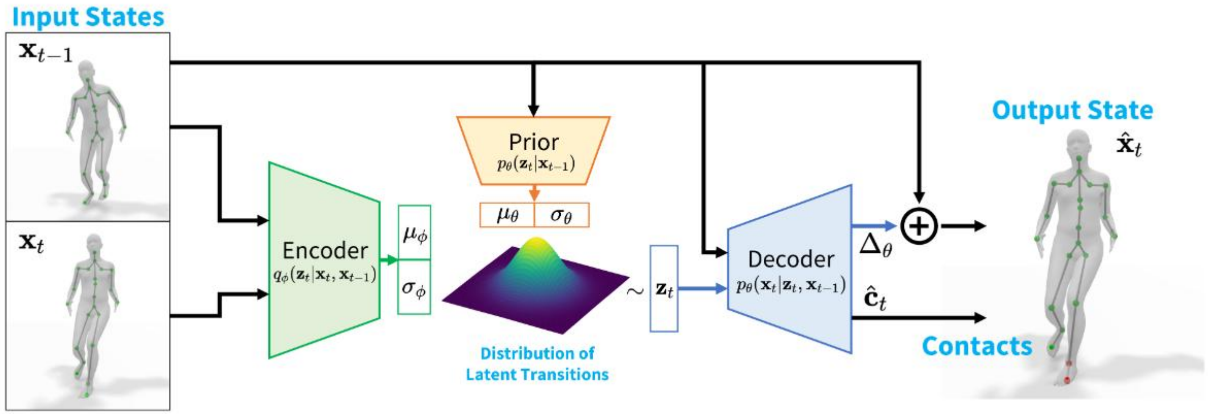
- Current methods
  - VPoser
    - Variational Human Pose Prior of SMPL-X
    - VAE with normal distribution

Background

Current methods

HuMoR

- Generative sampling tasks
- recover plausible pose sequences with noise and occlusions
- Estimation from 3D and RGB Observations



Background

- Other perspective:
  - To model the full manifold of plausible poses in high-dimensional pose space directly.
- implicit functions
  - a fully differentiable neural network
  - gradient descent

Outline

Method

- **Given a neural network:** $f : SO(3)^K \mapsto \mathbb{R}^+$
- **Represent the manifold of plausible poses as the zero level set:**

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

- **SMPL body model**

Method

1. Unit Quaternions as Representation of SO(3)

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

- SO(3)
- Distance between two poses:

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

Method

2. Hierarchical Implicit Neural Function

- Local coordinate frame ➡ continuous manipulation of a single joint
- ancestor rotations

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

Combined pose embedding: $\mathbf{p} = [\mathbf{v}_1 || \dots || \mathbf{v}_K]$

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

Method

3. Loss function

- **training data:** $\mathcal{D} = \{(\boldsymbol{\theta}_i, d_i)\}_{1 \le i \le N}$
- **standard distance loss:** $\mathcal{L}_{\text{UDF}} = \sum_{(\boldsymbol{\theta},d)\in\mathcal{D}} ||f^{\text{udf}}(\boldsymbol{\theta}) - d_{\boldsymbol{\theta}}||_2$
- **Eikonal regularizer (unit-norm gradient)** $\mathcal{L}_{\text{eikonal}} = \sum_{(\boldsymbol{\theta},d)\in\mathcal{D},\ d\neq 0} (||\nabla_{\boldsymbol{\theta}} f^{\text{udf}}(\boldsymbol{\theta})|| - 1)^2$

Method

4. Projection Algorithm

- converge to local minima on the sphere, assuming a correctly learned distance function, is the nearest point on the pose manifold.

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

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

Outline

Experiments

- **Set up:**
  - **Datasets: AMASS(Training, dθ = 0)**
  - **negative samples: distance dθ > 0**
  - **Training scheme: Increase the number of non-manifold poses with a small distance in each training batch.**
  - f^{enc} **: 2-layer MLP,** f^{df} **: 5-layer-MLP**

Experiments

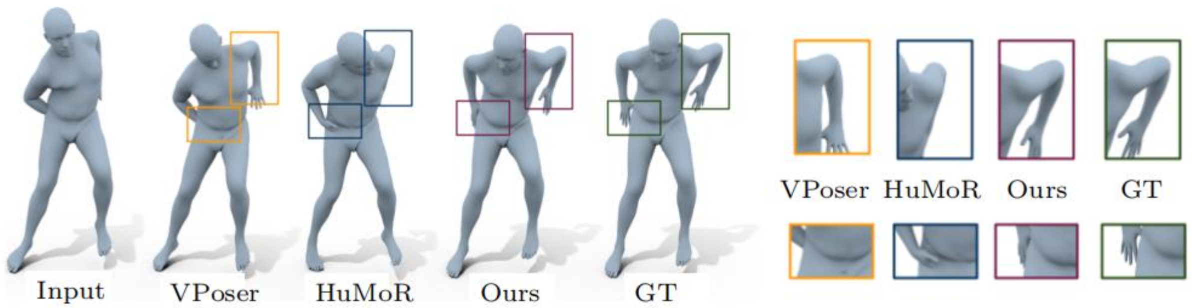
- Different uses:
  - Denoising Mocap Data
  - 3D pose Estimation from Images
  - Pose Generation
  - Pose Interpolation

Experiments

Denoising Mocap Data

1. motion denosing

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**



- Evaluate on two different settings:
  - clean (HPS, AMASS)
  - nosiy (add Gaussian noise to AMASS)
- Metrics: per-vertex error

Experiments

Denoising Mocap Data

1. motion denosing

$$\hat{\boldsymbol{\theta}}^t = \arg\min_{\boldsymbol{\theta}} \lambda_{\text{v}}\mathcal{L}_{\text{v}} + \lambda_{\boldsymbol{\theta}}\mathcal{L}_{\boldsymbol{\theta}} + \lambda_t\mathcal{L}_t$$

- data term: $\mathcal{L}_{\text{v}} = ||\mathcal{J}(\boldsymbol{\beta}_0, \hat{\boldsymbol{\theta}}^t) - \mathcal{J}_{\text{obs}}||_2^2$
- temporal smoothness term $\mathcal{L}_t = ||M(\boldsymbol{\beta}_0, \hat{\boldsymbol{\theta}}^{\text{t}}) - M(\boldsymbol{\beta}_0, \boldsymbol{\theta}^{\text{t}-1})||_2^2$
- proir term: $\mathcal{L}_{\boldsymbol{\theta}} = f^{\text{udf}}(\boldsymbol{\theta})$

Experiments

Denoising Mocap Data

2. Fitting to partial data (randomly create occluded poses)

- **Evaluate on three different type of occlusions: occluded left leg, occluded left arm and occluded right shoulder and upper arm**
- **Metrics: per-vertex error**

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

Experiments

3D pose Estimation from Images

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

- **data term:** $\mathcal{L}_J = \sum_{i \in \text{joints}} \gamma_i w_i \rho(\Pi_K(R_\theta(J(\beta))) - J_{\text{est,i}})$
- **shape regularizer:** $\mathcal{L}_\beta = ||\beta||^2$
- **prior term:** $\mathcal{L}_\theta = f^{\text{udf}}(\theta)$
- **bending term:** $\mathcal{L}_\alpha = \sum_{i \in (\text{elbow,knees})} \exp(\theta_i)$

$\boxed{\lambda_\theta = w f^{\text{udf}}(\theta)}$

- **To ensure that if the pose is getting close to the manifold, the prior term is down-weighted, which results in faster convergence.**

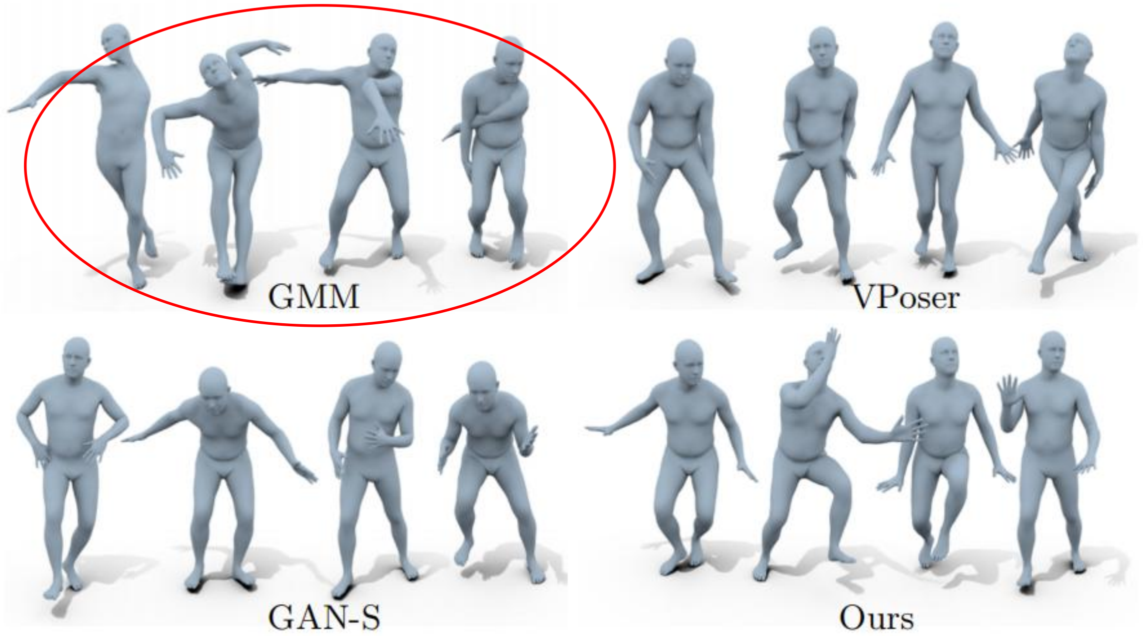
Experiments

- 3D pose Estimation from Images
  - EHF dataset
  - Refine the ExPose output with Pose-NDF as prior

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

Experiments

- **Pose Generation:**
- **Sample a random point from SO(3)K and project it onto the manifold.**



Experiments

Pose Generation

- Average Pairwise Distance (APD)： mean joint distance between all pairs of samples
  - the diversity of generated poses
- the percentage of self-intersecting faces in generated poses

	GMM	VPoser	GAN-S	Pose-NDF
APD	48.24	23.13	27.52	32.31
percentage	/	0.89%	1.43%	2.10%

Experiments

- **Pose Interpolation** $\theta_t = \theta'_{t-1} + \tau(\theta'_T - \theta'_{t-1})$
  - **Evaluate: mean per-vertex distance**
  - **Pose-NDF(2.72 ± 2.16), GAN-S (2.71 ± 2.45) , VPoser(2.53 ±4.62)**

Experiments

- **Pose-NDF vs. Gaussian Assumption models**
  - **the cumulative error based on deviation from the mean pose**
  - **AMASS Noisy (60 and 120 frames)**

	Pose-NDF	VPoser	HuMoR
σ	8.18	8.35	10.08
2σ	8.20	9.11	11.38
3σ	8.21	9.13	16.86

Outline

Conclusion

- human pose prior model
- a scalar neural distance
- zero level set in SO(3)K.
- Application:
  - diverse pose sampling
  - pose estimation from images
  - motion denoising.

Thanks!