3D Vision and
Robotics Lab

[CVPR 2022] BANMo: Building Animatable 3D Neural Models from Many Casual Videos

Gengshan Yang[2*] Minh Vo[3] Natalia Neverova[1] Deva Ramanan[2] Andrea Vedaldi[1] Hanbyul Joo[1]

[1]Meta AI [2]Carnegie Mellon University [3]Meta Reality Labs

Lab Seminar

Gyeongsu Cho @UNIST

2023.09.20. (Wed)

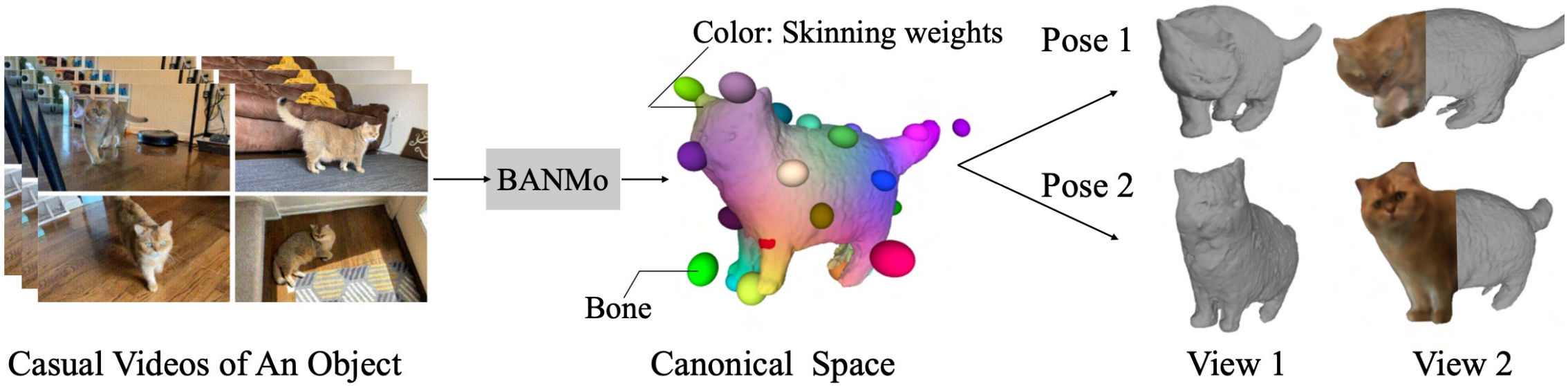
Contents

Introduction

Goal: Animatable 3D Models from Causal Videos

Introduction

Goal: Animatable 3D Models from Causal Videos

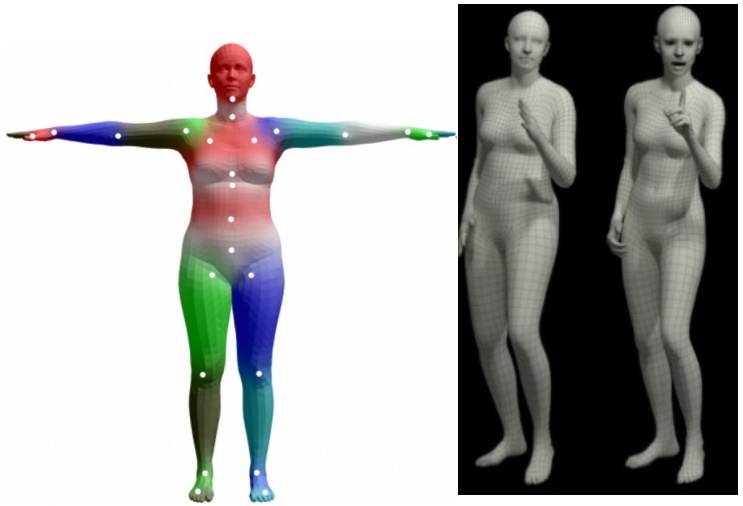


Introduction

Multi-view camera systems

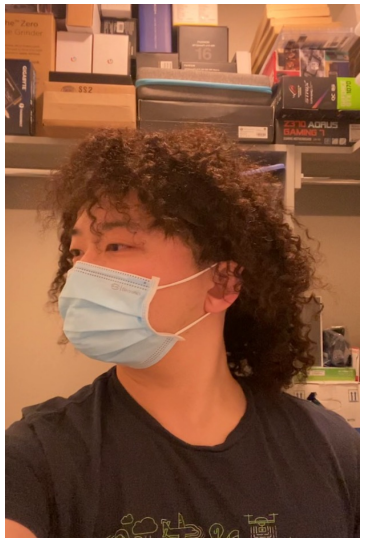
[CVPR 2021] Neural Body

Category-specific body models



[CVPR 2019] SMPL-X

3D Reconstruction
with differentiable rendering



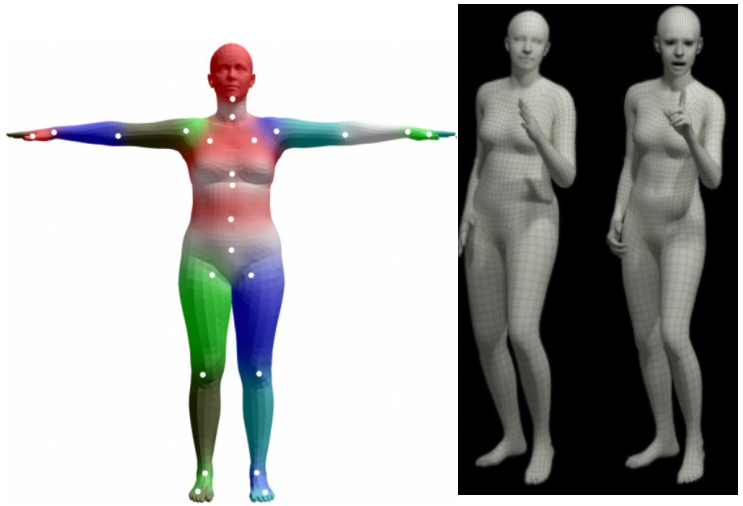
[ICCV 2021] Nerfies

Introduction

Multi-view camera systems

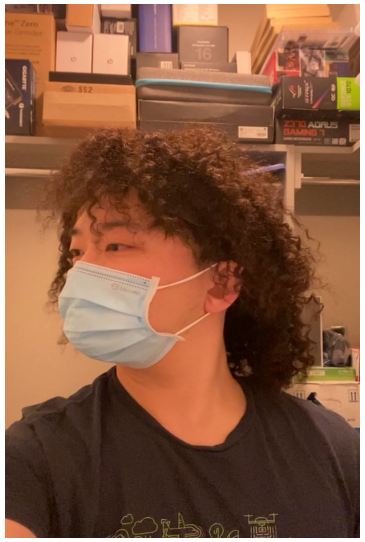
[CVPR 2021] Neural Body

Category-specific body models



[CVPR 2019] SMPL-X

3D Reconstruction
with differentiable rendering



[ICCV 2021] Nerfies

Monocular casual videos

Template-free

Freely-moving targets

Introduction

Introduction

1. How to represent 3D geometry and appearance of the target in a canonical space?

Introduction

1. How to represent 3D geometry and appearance of the target in a canonical space?

2. How to deform 3D points between canonical space and individual time instances?

Introduction

1. How to represent 3D geometry and appearance of the target in a canonical space?

2. How to deform 3D points between canonical space and individual time instances?

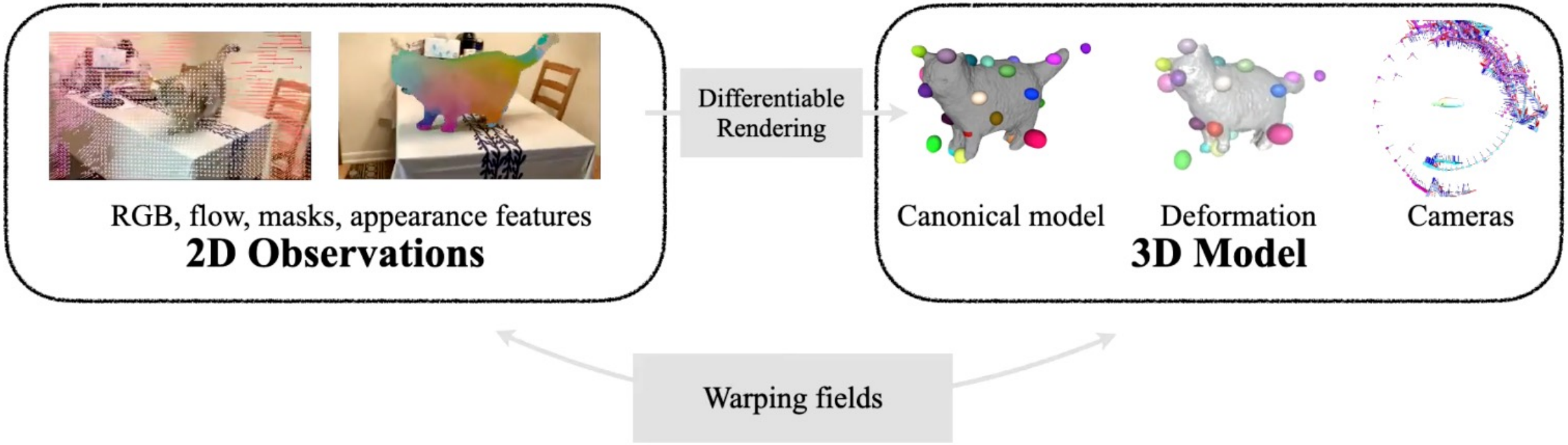
3. How to find pixel or part correspondences over videos given different viewpoint, lighting, background, and object deformations?

Introduction

1. How to represent 3D geometry and appearance of the target in a canonical space?
2. How to deform 3D points between canonical space and individual time instances?
3. How to find pixel or part correspondences over videos given different viewpoint, lighting, background, and object deformations?

Key Challenge : How to find the warping between the image and canonical 3D space?

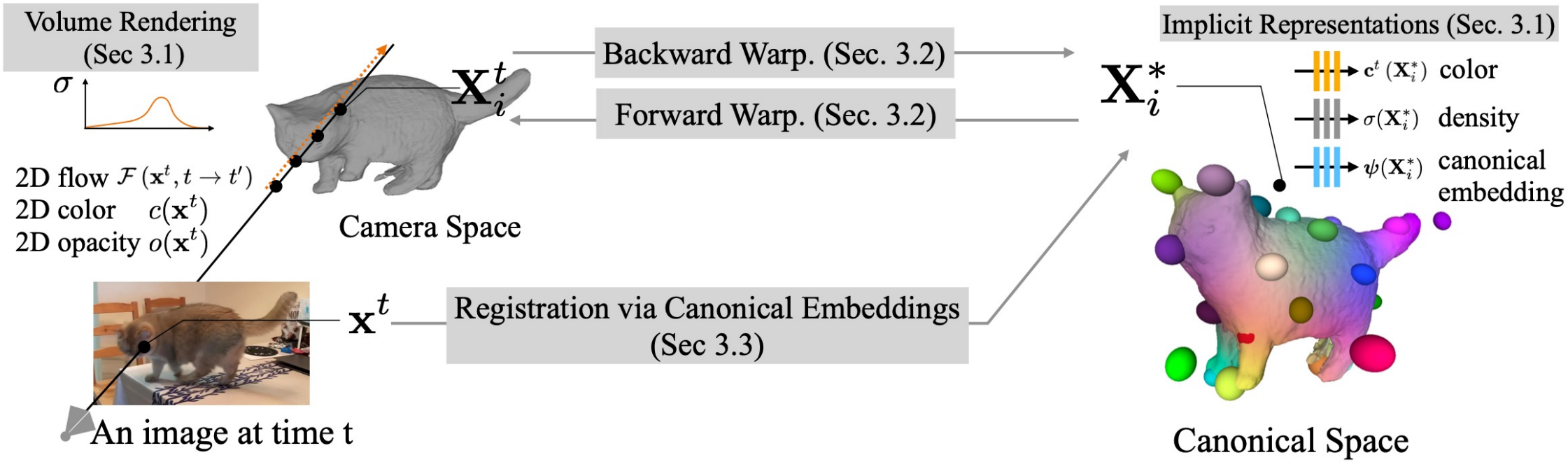
Introduction



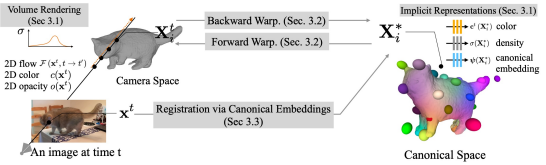
Key Challenge : How to find the warping between the image and canonical 3D space?

Method

Method


Volume Rendering (Sec 3.1)
σ
2D flow $\mathcal{F}(\mathbf{x}^t, t \rightarrow t')$
2D color $c(\mathbf{x}^t)$
2D opacity $o(\mathbf{x}^t)$
$\mathbf{X}_i^t$
Camera Space
Backward Warp. (Sec. 3.2)
Forward Warp. (Sec. 3.2)
$\mathbf{x}^t$
An image at time t
Registration via Canonical Embeddings (Sec 3.3)
Implicit Representations (Sec. 3.1)
$\mathbf{X}_i^*$
$\mathbf{c}^t(\mathbf{X}_i^*)$ color
$\sigma(\mathbf{X}_i^*)$ density
$\psi(\mathbf{X}_i^*)$ canonical embedding
Canonical Space

Method



1. Implicit 3D Representation

2. Differentiable Volume Rendering

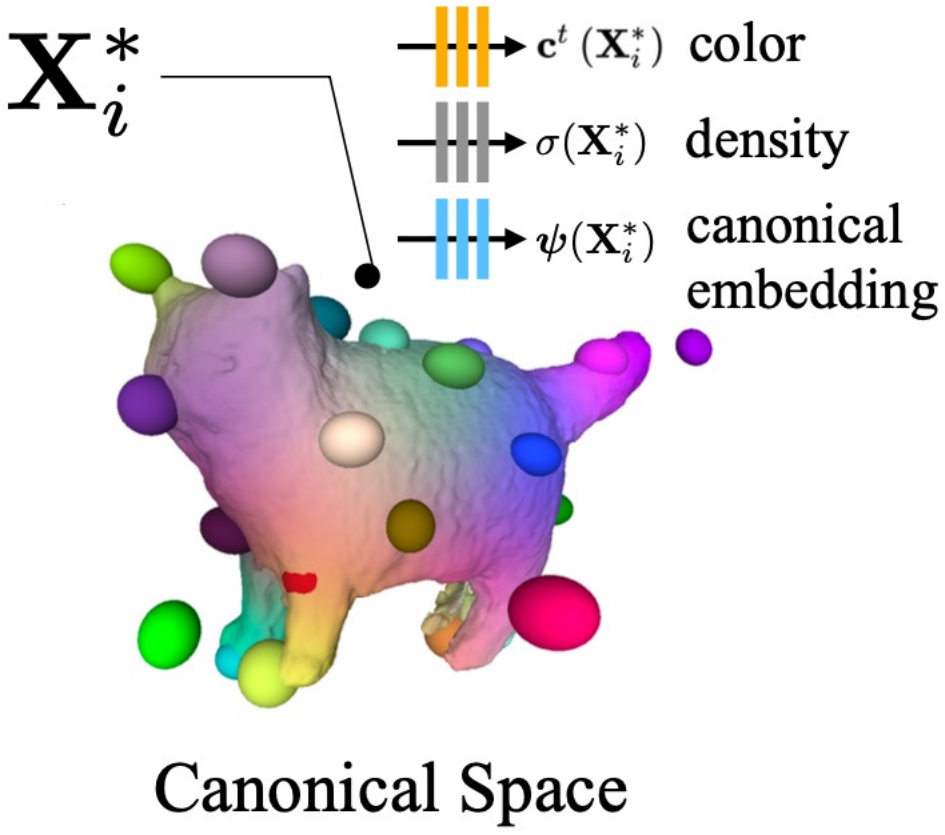
3. Deformation Model via Neural Blend Skinning

4. Registration

5. Optimization

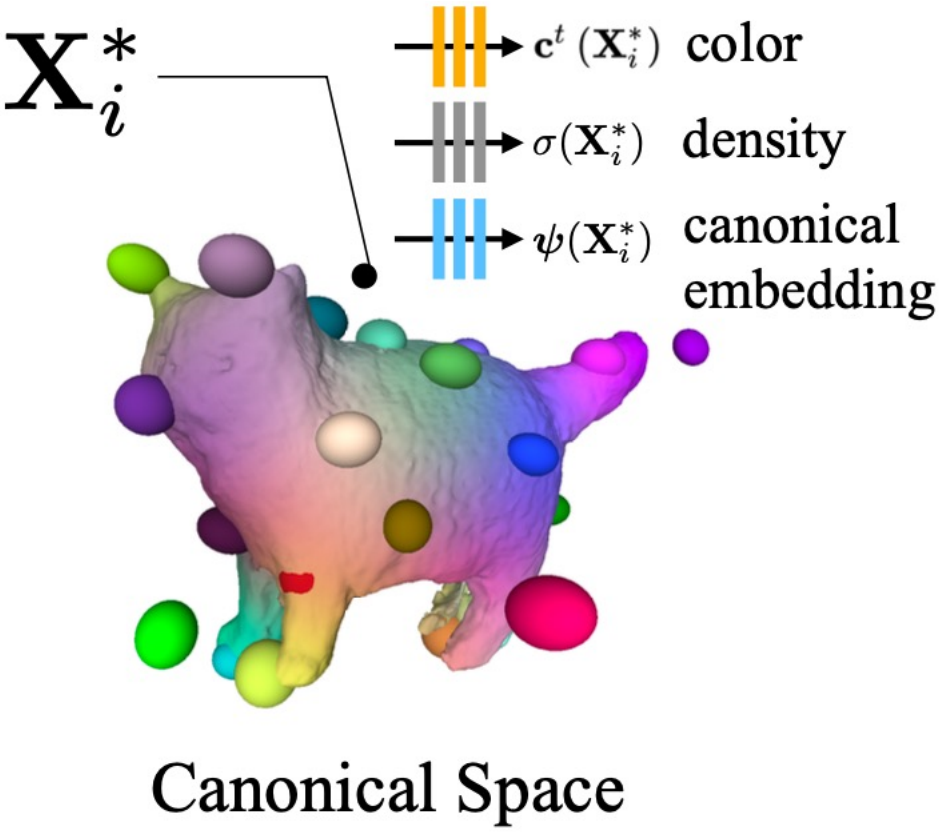
Method

1. Implicit 3D Representation



Method

1. Implicit 3D Representation



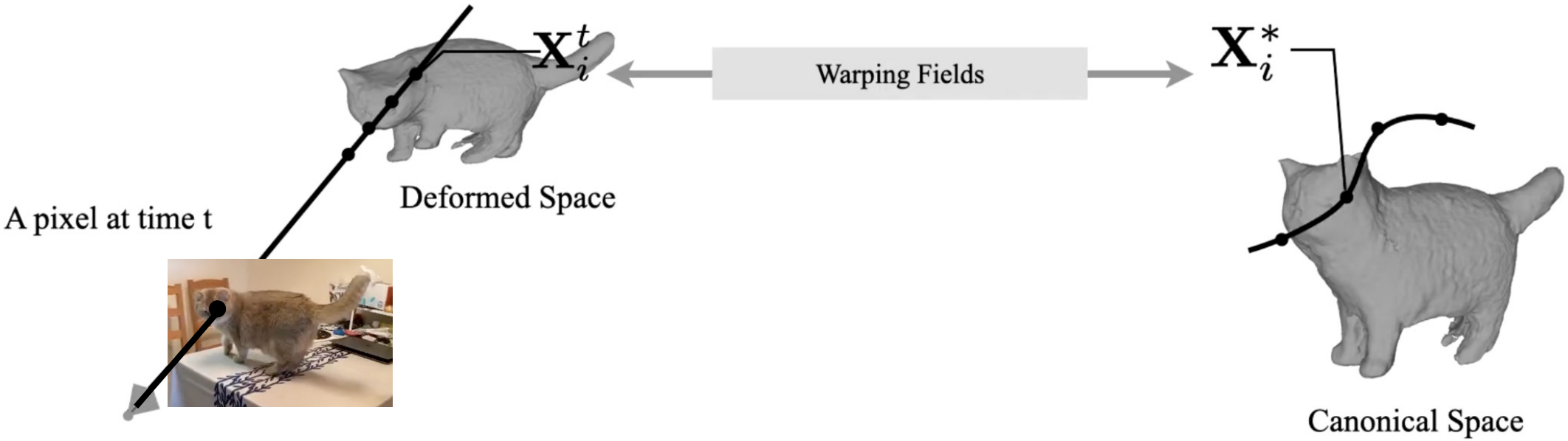
$$\mathbf{c}^t = \mathbf{MLP}_{\mathbf{c}}(\mathbf{X}^*, \mathbf{v}^t, \boldsymbol{\omega}_e^t), \quad (1)$$

$$\sigma = \Gamma_\beta(\mathbf{MLP}_{\mathrm{SDF}}(\mathbf{X}^*)), \quad (2)$$

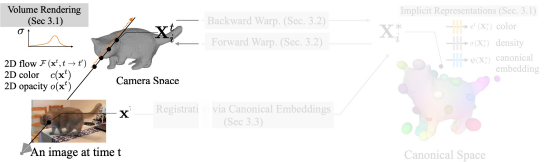
$$\boldsymbol{\psi} = \mathbf{MLP}_{\boldsymbol{\psi}}(\mathbf{X}^*). \quad (3)$$

Method

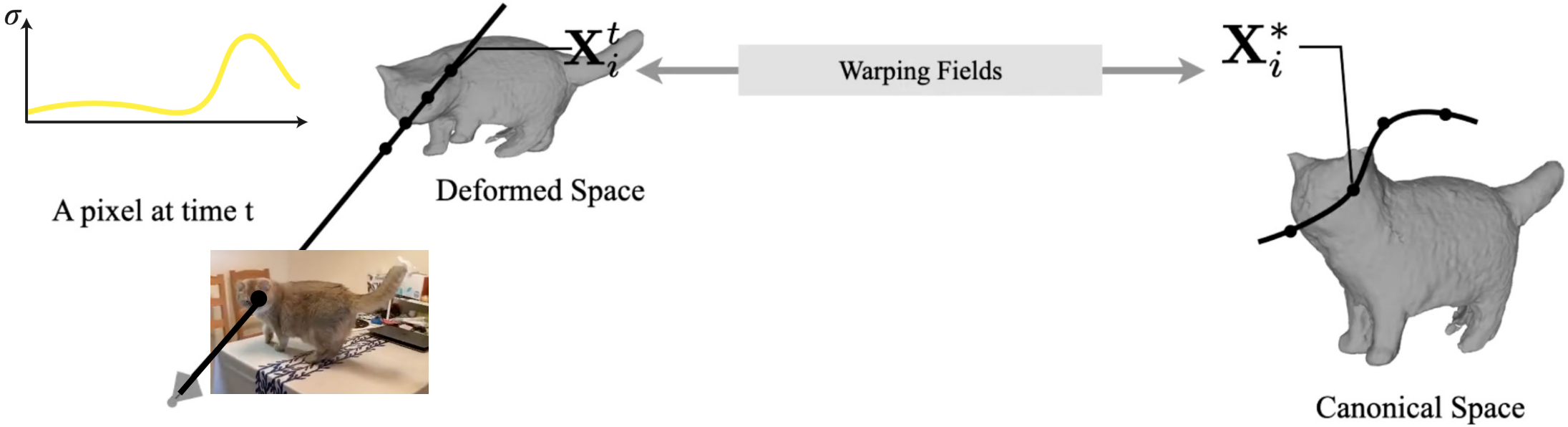
2. Differentiable Volume Rendering



Method



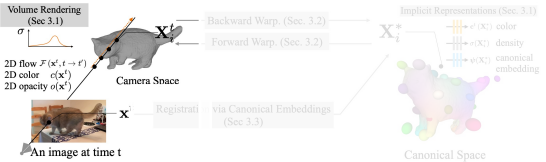
2. Differentiable Volume Rendering



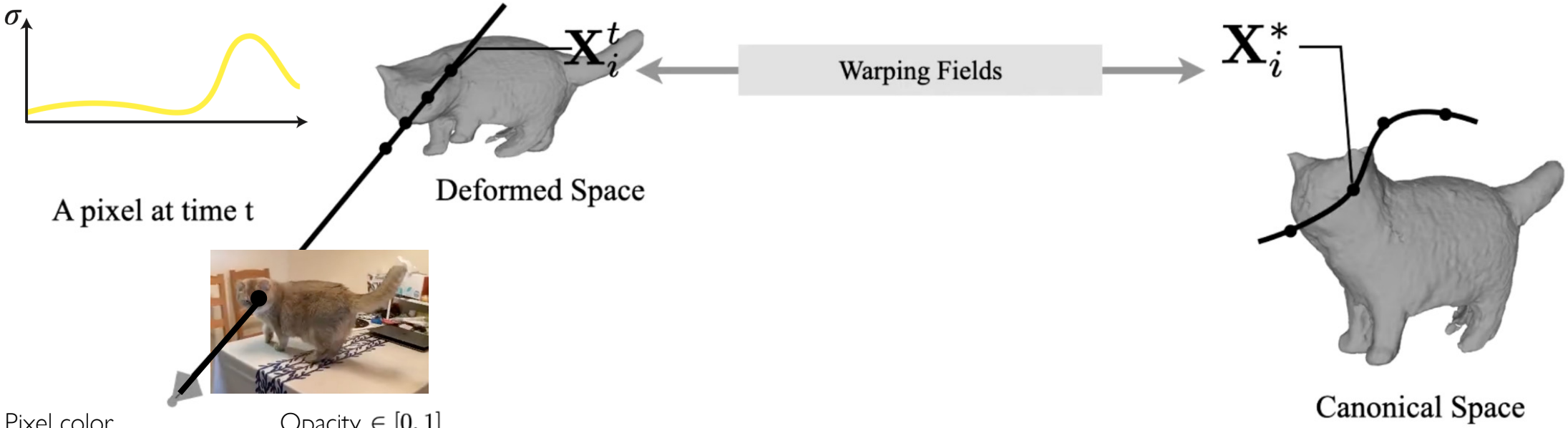
$$\mathbf{c}(\mathbf{x}^t) = \sum_{i}^{N} \tau_i \mathbf{c}_i^t, \quad o(\mathbf{x}^t) = \sum_{i}^{N} \tau_i, \tag{4}$$

$$\mathbf{X}^*(\mathbf{x}^t) = \sum_{i=1}^{N} \tau_i \mathbf{X}_i^*. \tag{5}$$

Method



2. Differentiable Volume Rendering



Pixel color

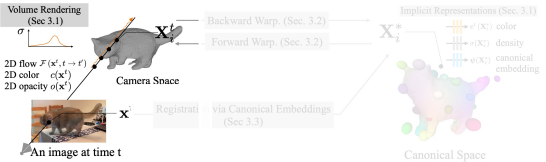
Opacity $\in [0, 1]$

$$\mathbf{c}(\mathbf{x}^t) = \sum_{i=1}^{N} \tau_i \mathbf{c}_i^t, \quad o(\mathbf{x}^t) = \sum_{i=1}^{N} \tau_i, \tag{4}$$

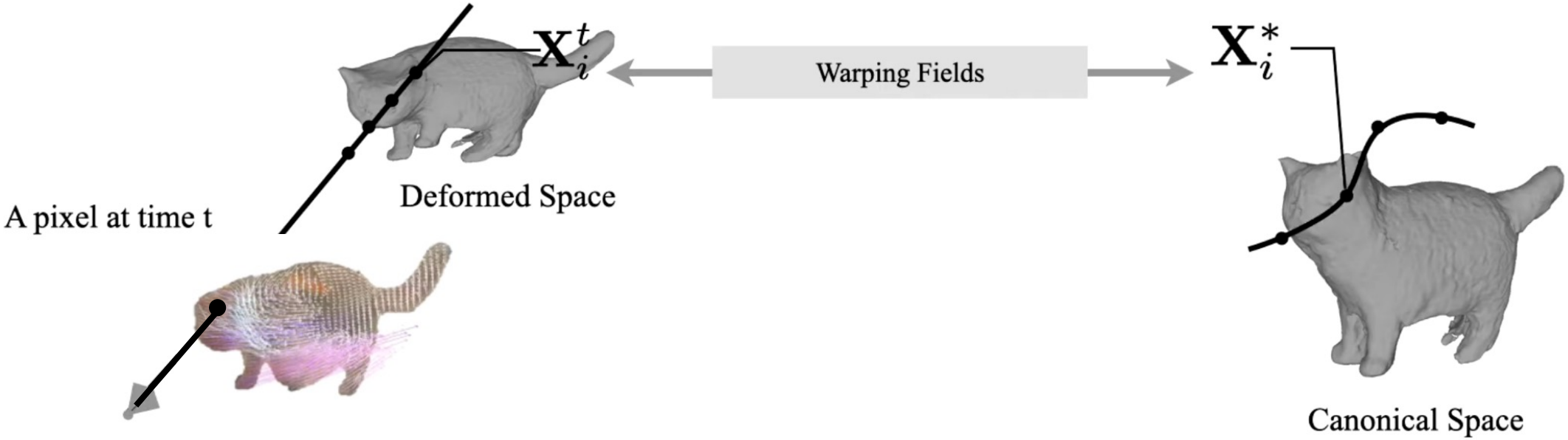
$$\mathbf{X}^*(\mathbf{x}^t) = \sum_{i=1}^{N} \tau_i \mathbf{X}_i^*. \tag{5}$$

expected surface intersection

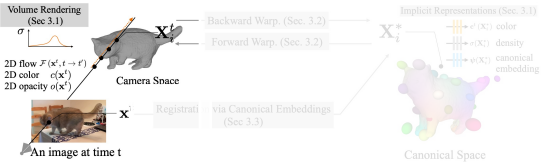
Method



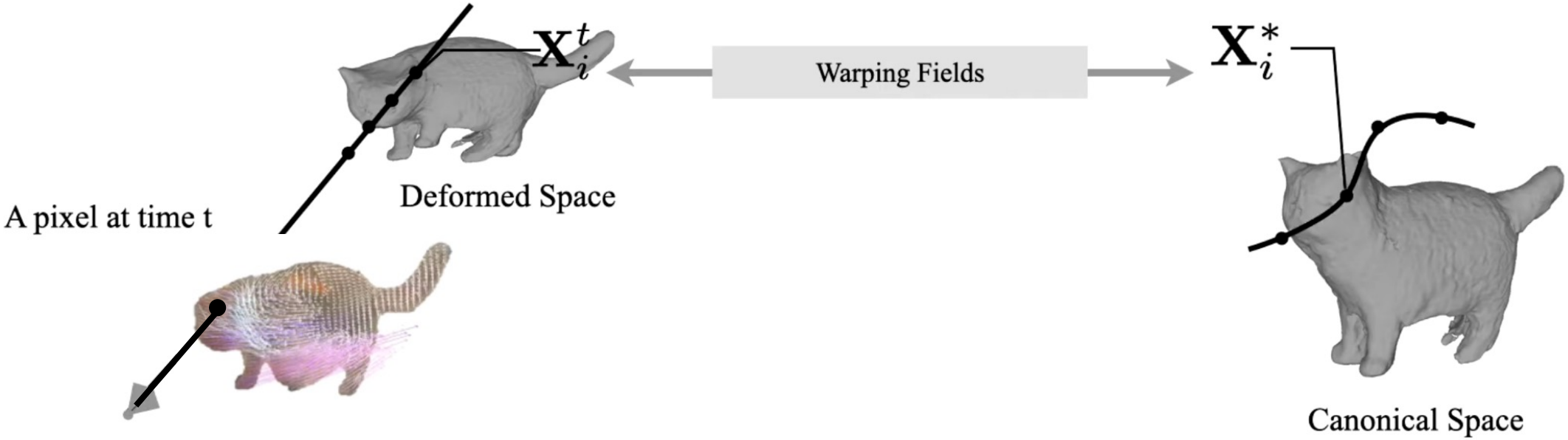
2. Differentiable Volume Rendering



Method



2. Differentiable Volume Rendering



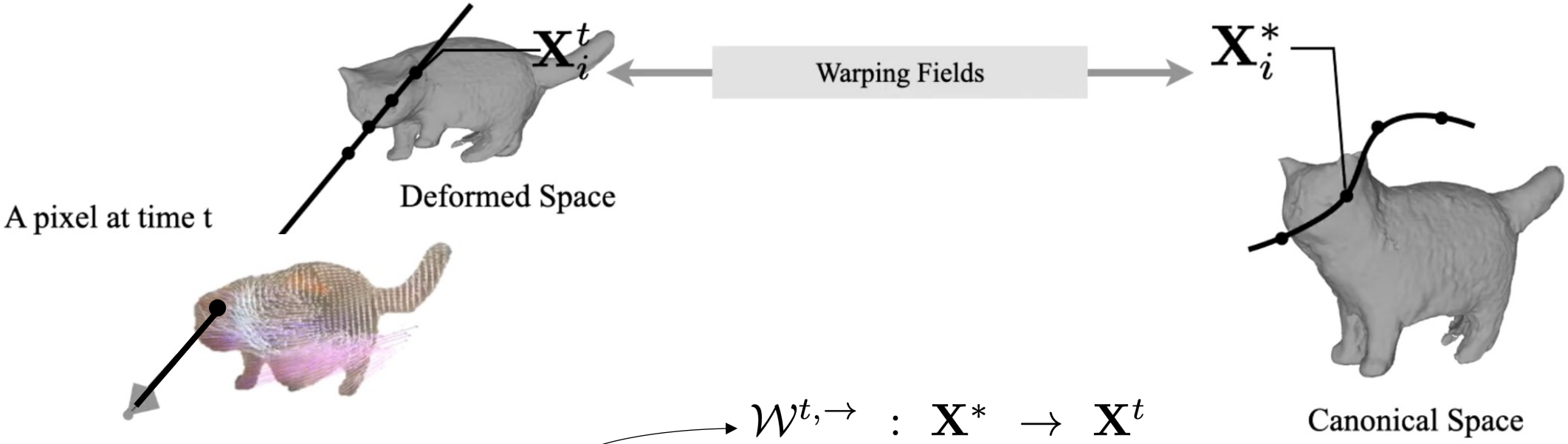
$$\mathbf{x}^{t'} = \sum_{i=1}^{N} \tau_i \Pi^{t'} \left(\mathcal{W}^{t',\rightarrow} (\mathbf{X}_i^*) \right), \qquad (6)$$

$$\mathcal{F} \left(\mathbf{x}^t, t \rightarrow t' \right) = \mathbf{x}^{t'} - \mathbf{x}^t. \qquad (7)$$

for optical flow loss

Method

2. Differentiable Volume Rendering



$$\mathcal{W}^{t,\rightarrow} : \mathbf{X}^* \rightarrow \mathbf{X}^t$$

expected 2D re-projection

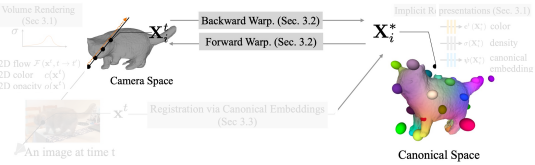
$$\mathbf{x}^{t'} = \sum_{i=1}^{N} \tau_i \Pi^{t'} \left(\mathcal{W}^{t',\rightarrow} (\mathbf{X}_i^*) \right), \qquad (6)$$

$$\mathcal{F} \left(\mathbf{x}^t, t \rightarrow t' \right) = \mathbf{x}^{t'} - \mathbf{x}^t. \qquad (7)$$

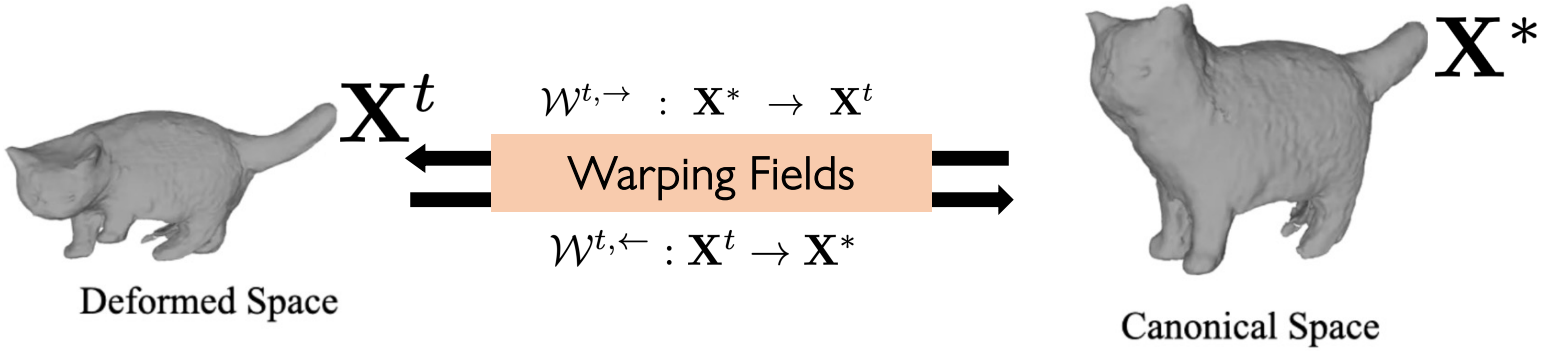
for optical flow loss

2D flow

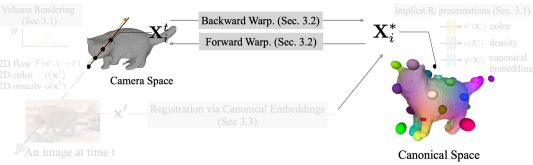
Method



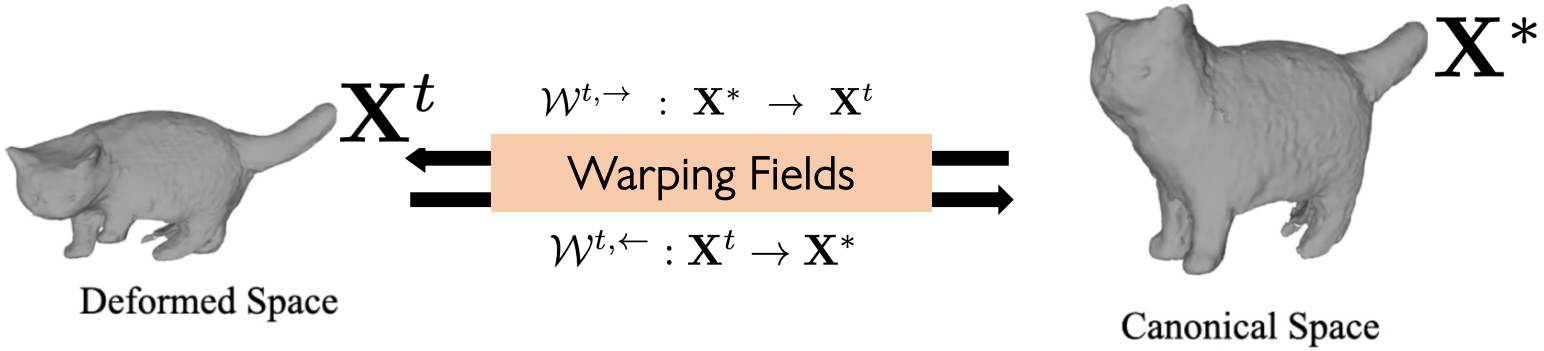
3. Deformation Model via Neural Blend Skinning



Method



3. Deformation Model via Neural Blend Skinning



$$\mathbf{X}^t = \mathcal{W}^{t,\rightarrow}(\mathbf{X}^*) = \mathbf{G}^t\mathbf{J}^{t,\rightarrow}\mathbf{X}^*, \quad (8)$$

$$\mathbf{X}^* = \mathcal{W}^{t,\leftarrow}(\mathbf{X}^t) = \mathbf{J}^{t,\leftarrow}(\mathbf{G}^t)^{-1}\mathbf{X}^t, \quad (9)$$

$$\mathbf{J}^{t,\rightarrow} = \sum_{b=1}^{B} \mathbf{W}_b^{t,\rightarrow}\mathbf{J}_b^t, \quad \mathbf{J}^{t,\leftarrow} = \sum_{b=1}^{B} \mathbf{W}_b^{t,\leftarrow}(\mathbf{J}_b^t)^{-1} \quad (10)$$

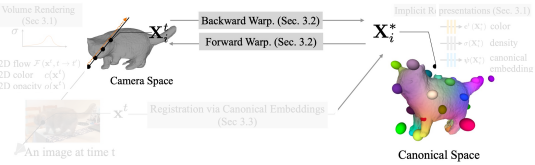
$$\mathbf{G}^t = \mathbf{MLP}_{\mathbf{G}}(\boldsymbol{\omega}_r^t), \quad \mathbf{J}_b^t = \mathbf{MLP}_{\mathbf{J}}(\boldsymbol{\omega}_b^t) \quad (11)$$

$$\omega_t^b = \mathbf{A}_i\mathcal{F}(t) \quad (12)$$

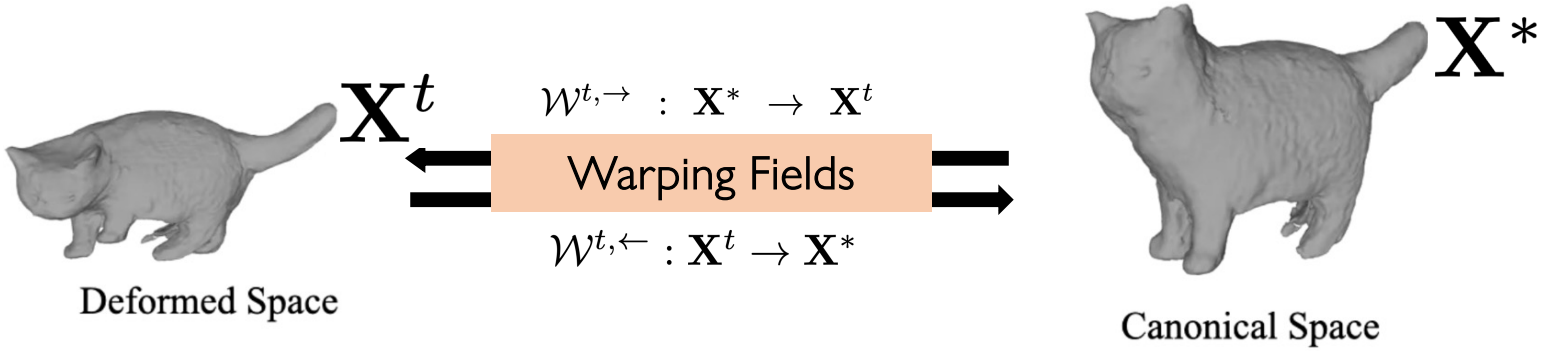
$$\mathbf{W}_\sigma = (\mathbf{X} - \mathbf{C}_b)^T\mathbf{Q}_b(\mathbf{X} - \mathbf{C}_b), \quad (13)$$

$$\mathbf{W} = \mathcal{S}(\mathbf{X}, \omega_b) = \sigma_{\text{softmax}}\left(\mathbf{W}_\sigma + \mathbf{W}_\Delta\right) \quad (14)$$

Method



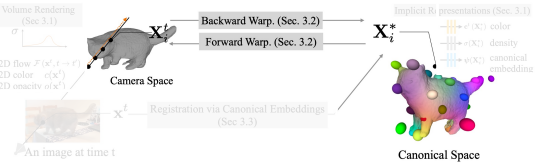
3. Deformation Model via Neural Blend Skinning



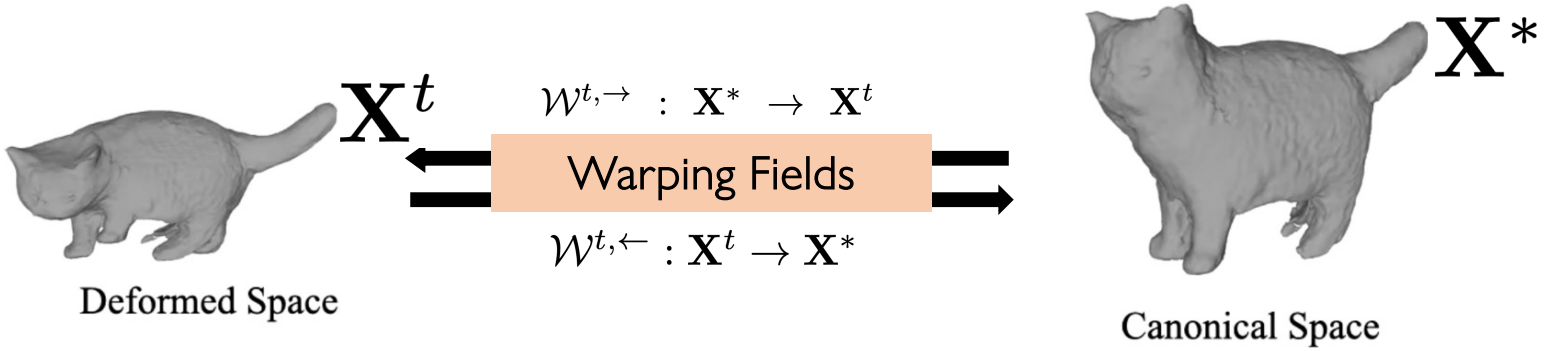
$$\mathbf{X}^t = \mathcal{W}^{t,\rightarrow}(\mathbf{X}^*) = \mathbf{G}^t\mathbf{J}^{t,\rightarrow}\mathbf{X}^*, \quad (8)$$

$$\mathbf{X}^* = \mathcal{W}^{t,\leftarrow}(\mathbf{X}^t) = \mathbf{J}^{t,\leftarrow}(\mathbf{G}^t)^{-1}\mathbf{X}^t, \quad (9)$$

Method



3. Deformation Model via Neural Blend Skinning

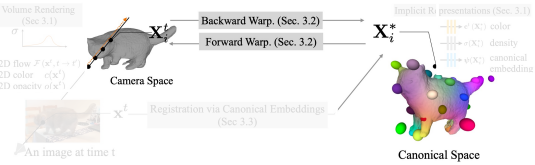


root body pose Bone(joint) pose

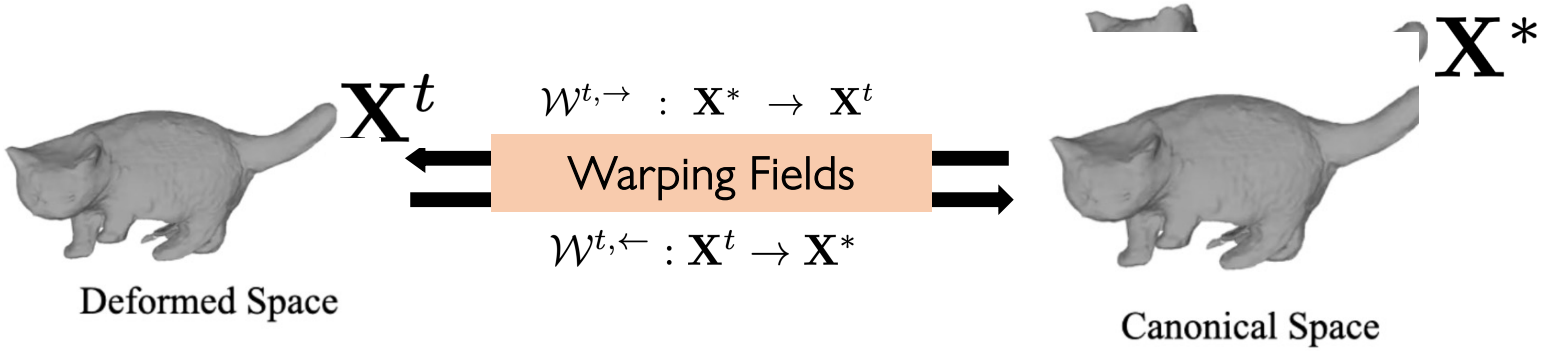
$$\mathbf{X}^t = \mathcal{W}^{t,\rightarrow}(\mathbf{X}^*) = \mathbf{G}^t \mathbf{J}^{t,\rightarrow} \mathbf{X}^*, \quad (8)$$

$$\mathbf{X}^* = \mathcal{W}^{t,\leftarrow}(\mathbf{X}^t) = \mathbf{J}^{t,\leftarrow} (\mathbf{G}^t)^{-1} \mathbf{X}^t, \quad (9)$$

Method



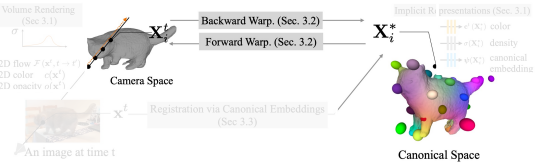
3. Deformation Model via Neural Blend Skinning



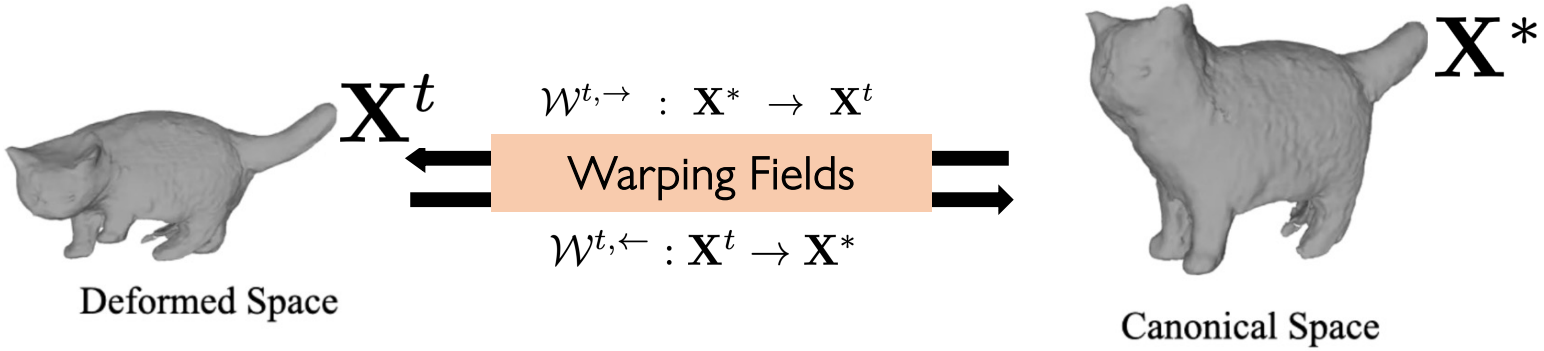
$$\mathbf{X}^t = \mathcal{W}^{t,\rightarrow}(\mathbf{X}^*) = \mathbf{G}^t \mathbf{J}^{t,\rightarrow} \mathbf{X}^*, \quad (8)$$

$$\mathbf{X}^* = \mathcal{W}^{t,\leftarrow}(\mathbf{X}^t) = \mathbf{J}^{t,\leftarrow}(\mathbf{G}^t)^{-1} \mathbf{X}^t, \quad (9)$$

Method



3. Deformation Model via Neural Blend Skinning



root body pose Bone(joint) pose

$$\mathbf{X}^t = \mathcal{W}^{t,\rightarrow}(\mathbf{X}^*) = \mathbf{G}^t\mathbf{J}^{t,\rightarrow}\mathbf{X}^*, \quad (8)$$

$$\mathbf{X}^* = \mathcal{W}^{t,\leftarrow}(\mathbf{X}^t) = \mathbf{J}^{t,\leftarrow}(\mathbf{G}^t)^{-1}\mathbf{X}^t, \quad (9)$$

$$\mathbf{J}^{t,\rightarrow} = \sum_{b=1}^{B} \mathbf{W}_b^{t,\rightarrow}\mathbf{J}_b^t, \quad \mathbf{J}^{t,\leftarrow} = \sum_{b=1}^{B} \mathbf{W}_b^{t,\leftarrow}(\mathbf{J}_b^t)^{-1} \quad (10)$$

$$\mathbf{G}^t = \mathbf{MLP}_{\mathbf{G}}(\boldsymbol{\omega}_r^t), \quad \mathbf{J}_b^t = \mathbf{MLP}_{\mathbf{J}}(\boldsymbol{\omega}_b^t) \quad (11)$$

$$\omega_t^b = \mathbf{A}_i\mathcal{F}(t) \quad (12)$$

latent code

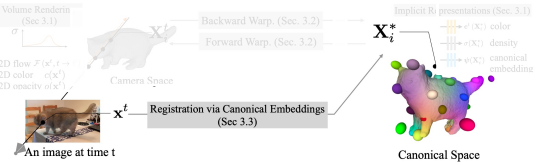
$$\mathbf{W}_\sigma = (\mathbf{X} - \mathbf{C}_b)^T\mathbf{Q}_b(\mathbf{X} - \mathbf{C}_b), \quad (13)$$

Skining weights

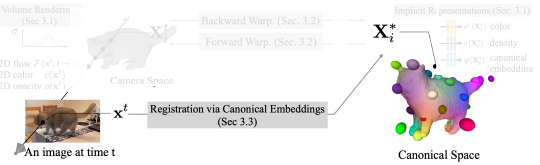
$$\mathbf{W} = \mathcal{S}(\mathbf{X}, \omega_b) = \sigma_{\mathrm{softmax}}\left(\mathbf{W}_\sigma + \mathbf{W}_\Delta\right) \quad (14)$$

Method

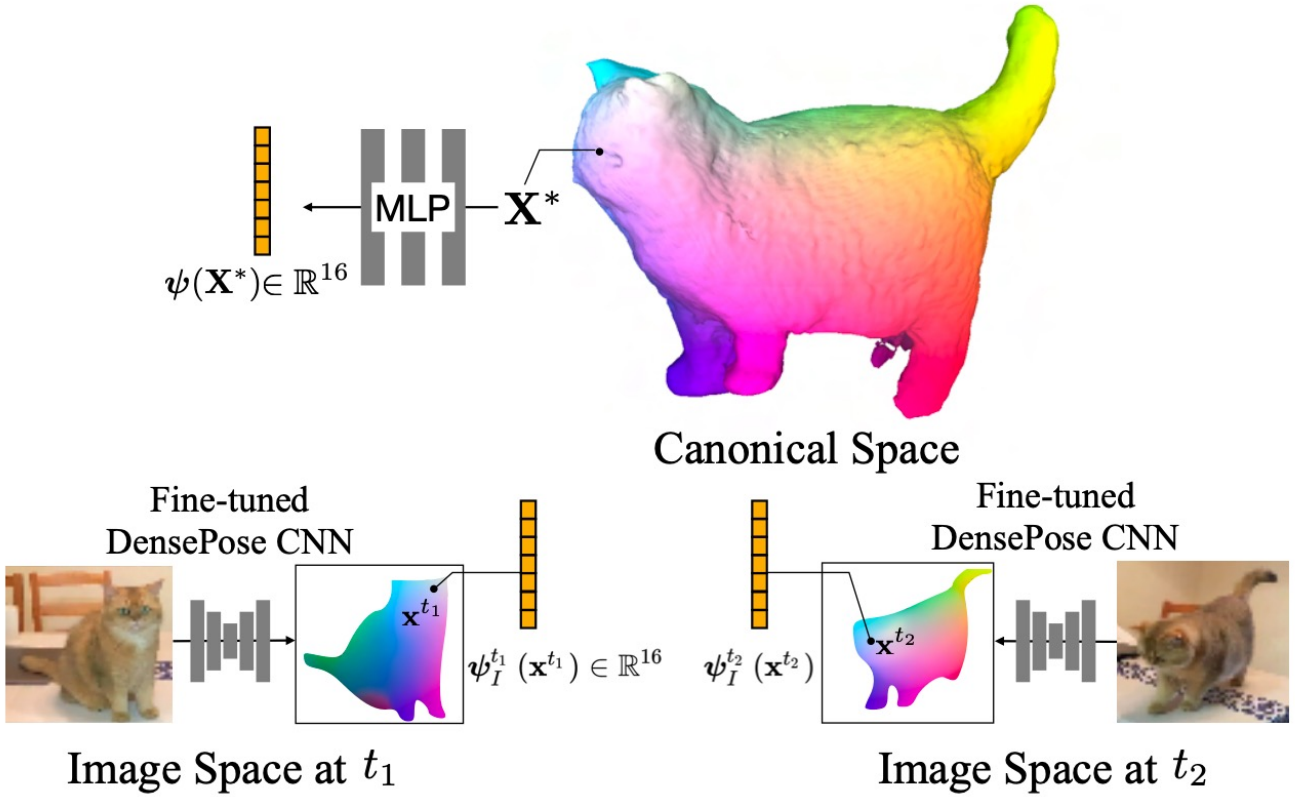
4. Registration



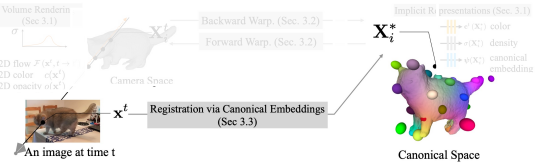
Method



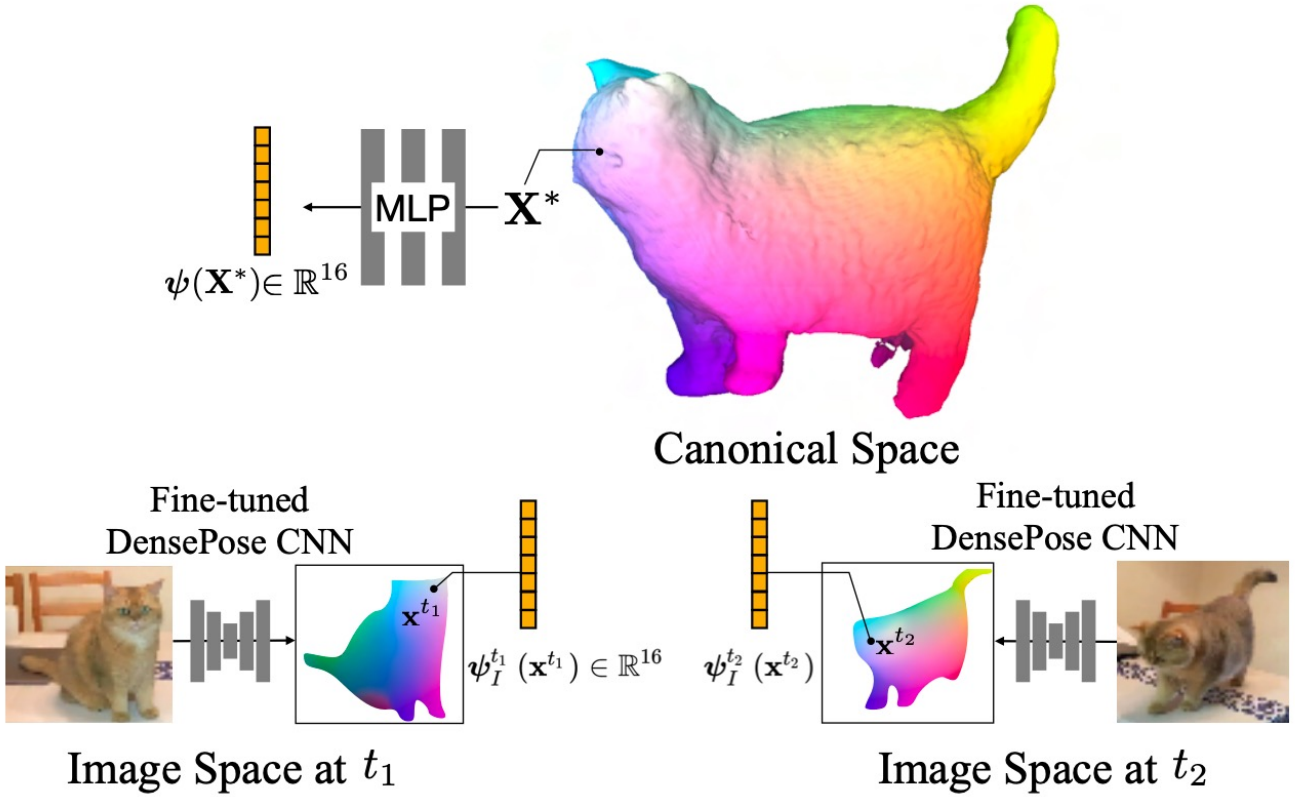
4. Registration



Method

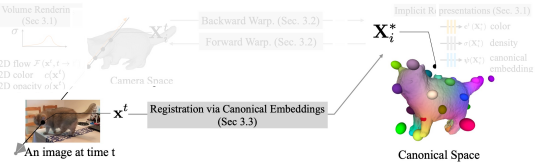


4. Registration

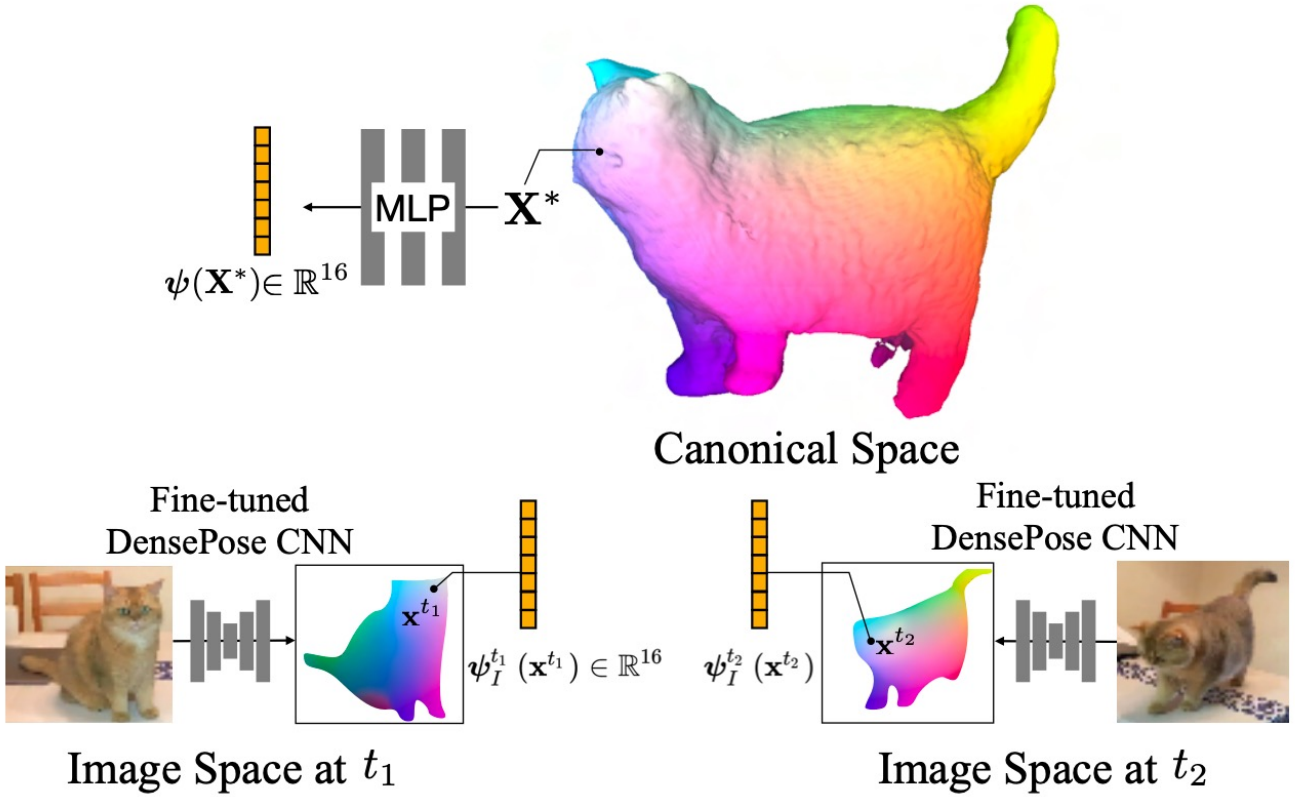


$$\hat{\mathbf{X}}^*(\mathbf{x}^t) = \sum_{\mathbf{X} \in \mathbf{V}^*} \tilde{\mathbf{s}}^t(\mathbf{x}^t)\mathbf{X}, \qquad (15)$$

Method



4. Registration



$$\hat{\mathbf{X}}^*(\mathbf{x}^t) = \sum_{\mathbf{X} \in \mathbf{V}^*} \tilde{\mathbf{s}}^t(\mathbf{x}^t)\mathbf{X}, \tag{15}$$

$$\tilde{\mathbf{s}}^t(\mathbf{x}^t) = \sigma_{\text{softmax}}\left(\alpha_s \left\langle \psi_I^t(\mathbf{x}^t), \psi(\mathbf{X}) \right\rangle\right)$$

Method

5. Optimization

1 $\{\mathbf{MLP_c},\ \mathbf{MLP_{SDF}},\ \mathbf{MLP}_{\boldsymbol{\psi}},\ \mathbf{MLP_G},\ \mathbf{MLP_J},\ \mathbf{MLP}_{\Delta}\}$

2 $\{\boldsymbol{\omega}_e^t, \boldsymbol{\omega}_r^t, \boldsymbol{\omega}_b^t, \boldsymbol{\omega}_b^*\}$

3 pixel embeddings $\boldsymbol{\psi}_I$

Method

5. Optimization

1 $\{\mathbf{MLP_c},\ \mathbf{MLP_{SDF}},\ \mathbf{MLP}_{\boldsymbol{\psi}},\ \mathbf{MLP_G},\ \mathbf{MLP_J},\ \mathbf{MLP}_{\Delta}\}$

2 $\{\boldsymbol{\omega}_e^t, \boldsymbol{\omega}_r^t, \boldsymbol{\omega}_b^t, \boldsymbol{\omega}_b^*\}$

3 pixel embeddings $\boldsymbol{\psi}_I$

Total Loss $$\mathcal{L} = \underbrace{\left(\mathcal{L}_{\text{sil}} + \mathcal{L}_{\text{rgb}} + \mathcal{L}_{\text{OF}}\right)}_{\text{reconstruction losses}} + \underbrace{\left(\mathcal{L}_{\text{match}} + \mathcal{L}_{\text{2D-cyc}}\right)}_{\text{feature registration losses}} + \mathcal{L}_{\text{3D-cyc}}.$$

Method

5. Optimization

Total Loss $$\mathcal{L} = \underbrace{\left(\mathcal{L}_{\text{sil}} + \mathcal{L}_{\text{rgb}} + \mathcal{L}_{\text{OF}}\right)}_{\text{reconstruction losses}} + \underbrace{\left(\mathcal{L}_{\text{match}} + \mathcal{L}_{\text{2D-cyc}}\right)}_{\text{feature registration losses}} + \mathcal{L}_{\text{3D-cyc}}.$$

Method

5. Optimization

Total Loss $\mathcal{L} = \underbrace{\left(\mathcal{L}_{\text{sil}} + \mathcal{L}_{\text{rgb}} + \mathcal{L}_{\text{OF}}\right)}_{\text{reconstruction losses}} + \underbrace{\left(\mathcal{L}_{\text{match}} + \mathcal{L}_{\text{2D-cyc}}\right)}_{\text{feature registration losses}} + \mathcal{L}_{\text{3D-cyc}}.$

$$\mathcal{L}_{\text{rgb}} = \sum_{\mathbf{x}^t} \left\| \mathbf{c}(\mathbf{x}^t) - \hat{\mathbf{c}}(\mathbf{x}^t) \right\|^2, \quad \mathcal{L}_{\text{sil}} = \sum_{\mathbf{x}^t} \left\| \mathbf{o}(\mathbf{x}^t) - \hat{\mathbf{s}}(\mathbf{x}^t) \right\|^2$$

$$\mathcal{L}_{\text{OF}} = \sum_{\mathbf{x}^t, (t, t')} \left\| \mathcal{F}\left(\mathbf{x}^t, t \rightarrow t'\right) - \hat{\mathcal{F}}\left(\mathbf{x}^t, t \rightarrow t'\right) \right\|^2, \quad (16)$$

Method

5. Optimization

Total Loss $\mathcal{L} = \underbrace{\left(\mathcal{L}_{\text{sil}} + \mathcal{L}_{\text{rgb}} + \mathcal{L}_{\text{OF}}\right)}_{\text{reconstruction losses}} + \underbrace{\left(\mathcal{L}_{\text{match}} + \mathcal{L}_{\text{2D-cyc}}\right)}_{\text{feature registration losses}} + \mathcal{L}_{\text{3D-cyc}}.$

$$\mathcal{L}_{\text{match}} = \sum_{\mathbf{x}^t} \left\| \hat{\mathbf{X}}^*(\mathbf{x}^t) - \mathbf{X}^*(\mathbf{x}^t) \right\|_2^2, \qquad (17)$$

$$\mathcal{L}_{\text{2D-cyc}} = \sum_{\mathbf{x}^t} \left\| \Pi^t \left(\mathcal{W}^{t,\rightarrow}(\hat{\mathbf{X}}^*(\mathbf{x}^t)) \right) - \mathbf{x}^t \right\|_2^2. \qquad (18)$$

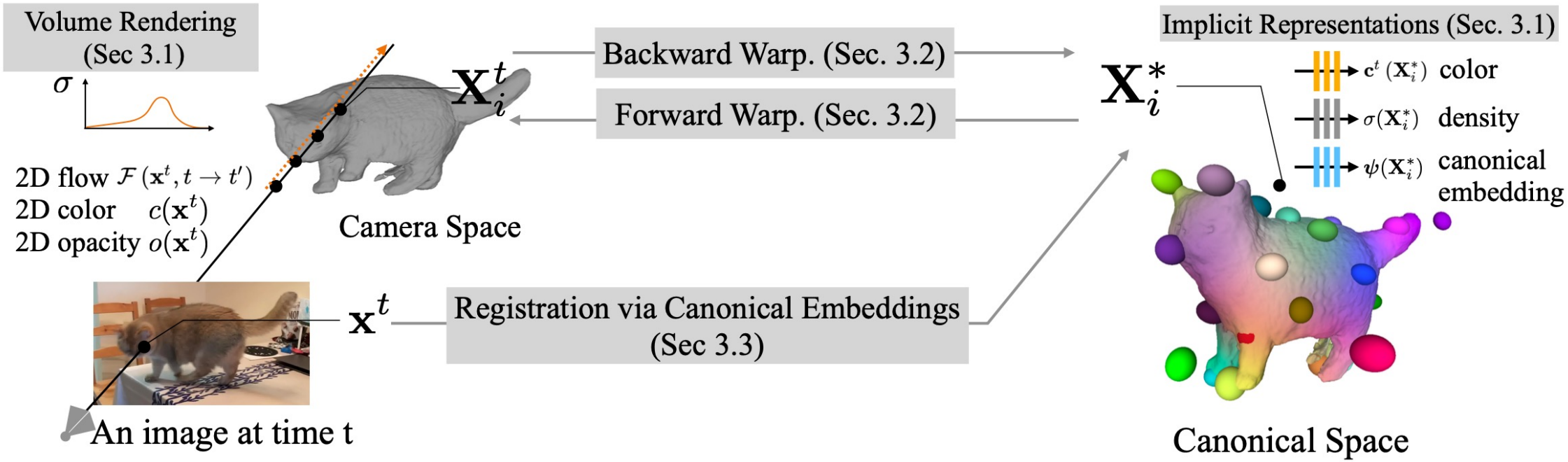
Method

5. Optimization

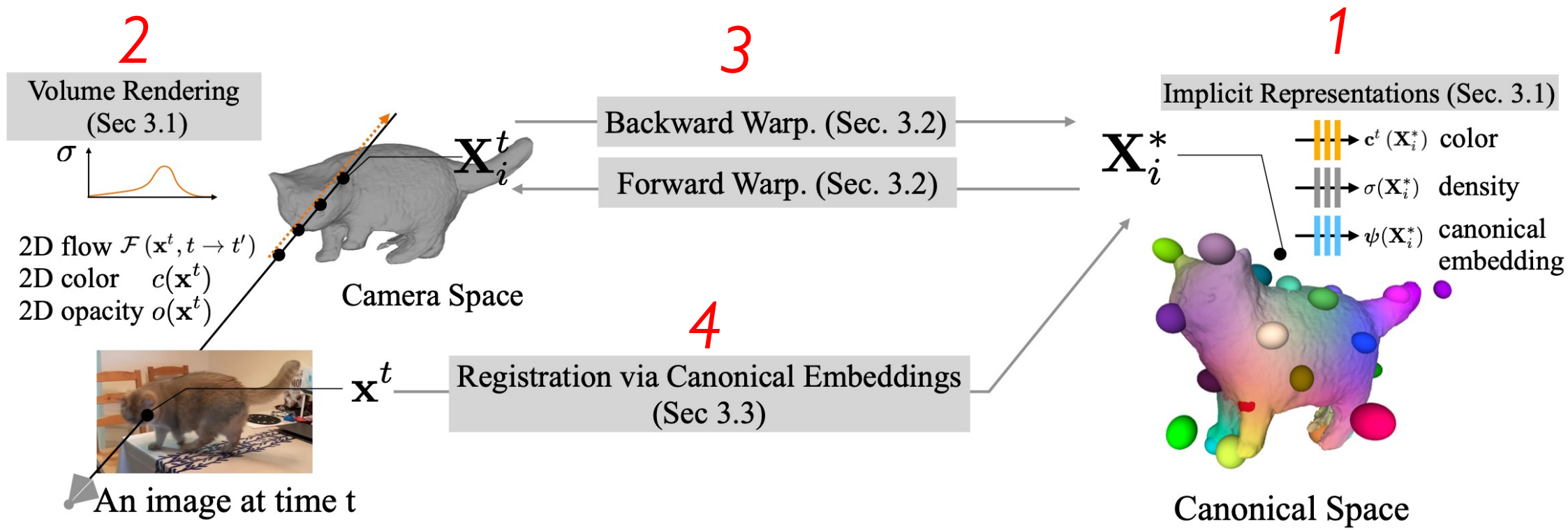
Total Loss $\mathcal{L} = \underbrace{\left(\mathcal{L}_{\text{sil}} + \mathcal{L}_{\text{rgb}} + \mathcal{L}_{\text{OF}}\right)}_{\text{reconstruction losses}} + \underbrace{\left(\mathcal{L}_{\text{match}} + \mathcal{L}_{\text{2D-cyc}}\right)}_{\text{feature registration losses}} + \mathcal{L}_{\text{3D-cyc}}.$

$$\mathcal{L}_{\text{3D-cyc}} = \sum_i \tau_i \left\| \mathcal{W}^{t,\rightarrow}\left(\mathcal{W}^{t,\leftarrow}(\mathbf{X}_i^t)\right) - \mathbf{X}_i^t \right\|_2^2, \quad (19)$$

Method


Volume Rendering (Sec 3.1)
σ
2D flow $\mathcal{F}(\mathbf{x}^t, t \rightarrow t')$
2D color $c(\mathbf{x}^t)$
2D opacity $o(\mathbf{x}^t)$
$\mathbf{X}_i^t$
Camera Space
Backward Warp. (Sec. 3.2)
Forward Warp. (Sec. 3.2)
$\mathbf{x}^t$
Registration via Canonical Embeddings (Sec 3.3)
An image at time t
Implicit Representations (Sec. 3.1)
$\mathbf{X}_i^*$
$\mathbf{c}^t(\mathbf{X}_i^*)$ color
$\sigma(\mathbf{X}_i^*)$ density
$\psi(\mathbf{X}_i^*)$ canonical embedding
Canonical Space

Method



5: Total Loss $\mathcal{L} = \underbrace{\left(\mathcal{L}_{\text{sil}} + \mathcal{L}_{\text{rgb}} + \mathcal{L}_{\text{OF}}\right)}_{\text{reconstruction losses}} + \underbrace{\left(\mathcal{L}_{\text{match}} + \mathcal{L}_{\text{2D-cyc}}\right)}_{\text{feature registration losses}} + \mathcal{L}_{\text{3D-cyc}}.$

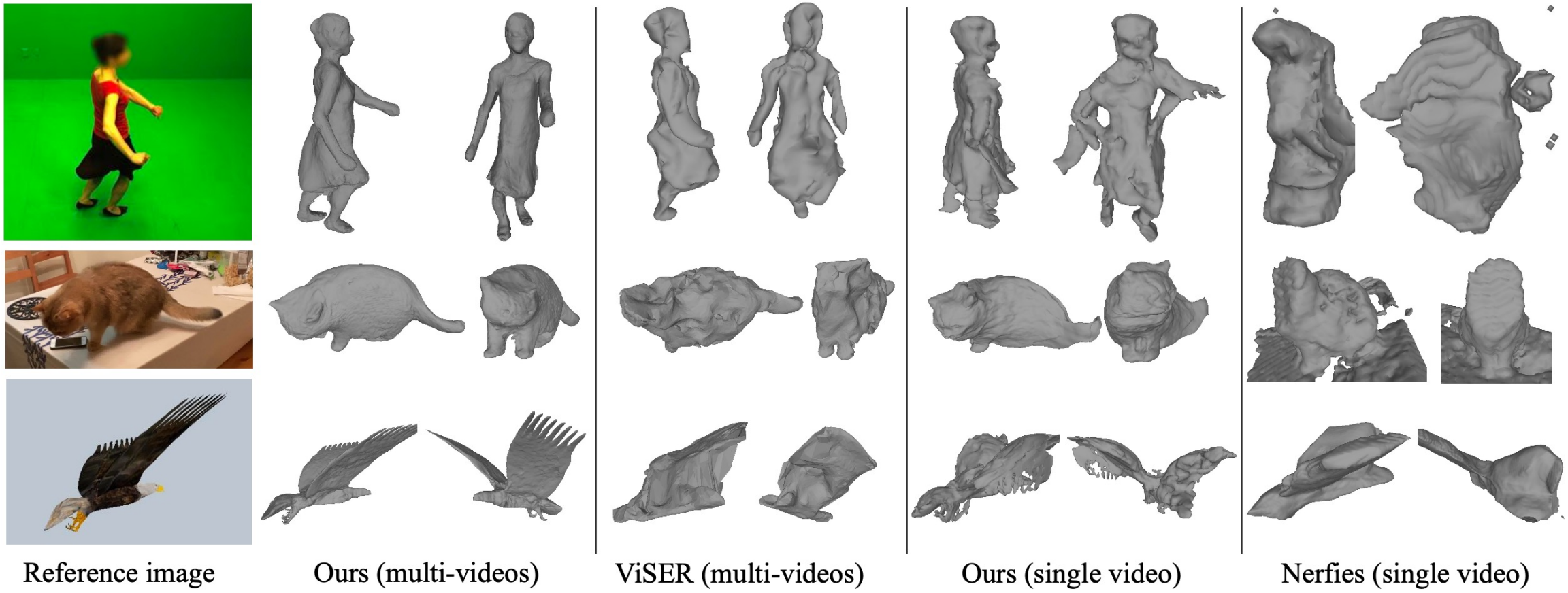
Experiments

- Dataset
  - Casual –videos dataset
  - AMA dataset
  - Animated Objects dataset

- Evaluation metrics
  - CD
  - F-Score

Experiments

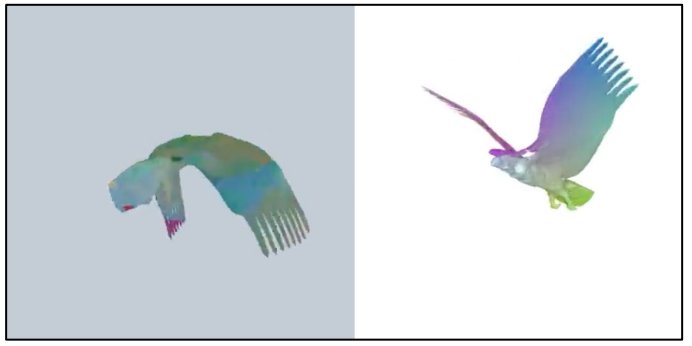
Qualitative comparison of our method with prior work

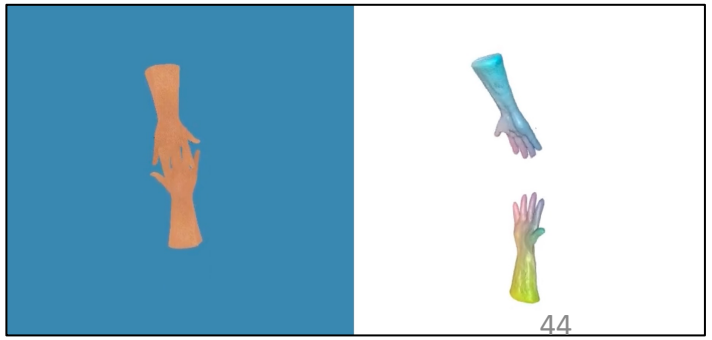


Experiments

Quantitative results on AMA and Animated Objects

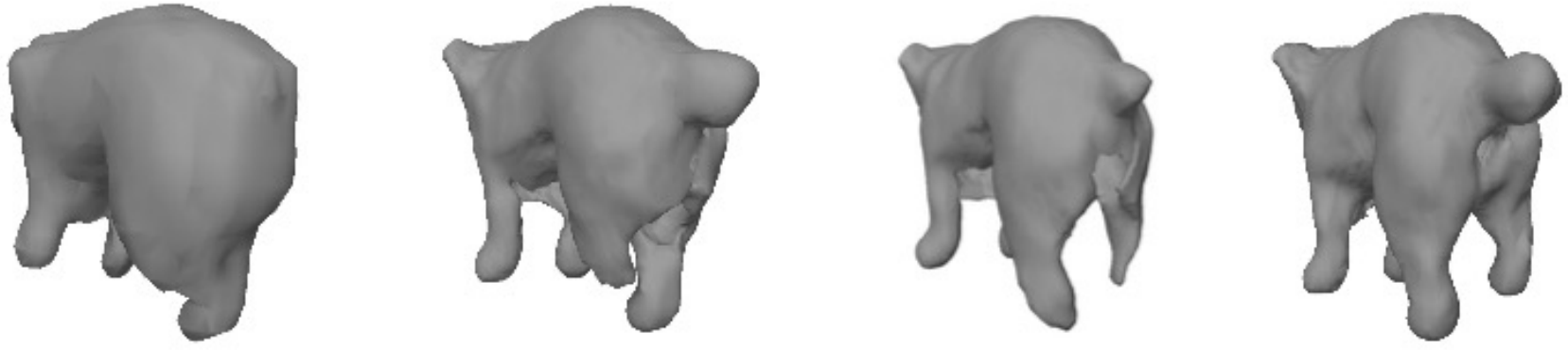
Method	AMA-swing		Eagle*		Hands*	
	CD	F@2%	CD	F@2%	CD	F@2%
Ours	**9.1**	**57.0**	**8.1**	**56.7**	**7.5**	**49.6**
ViSER	15.7	52.2	23.0	20.6	16.8	21.3
Ours^S	9.4	56.8	10.8	48.6	10.5	35.2
Nerfies^S	22.6	13.2	18.4	18.0	24.4	14.9





Experiments

Reconstruction Results

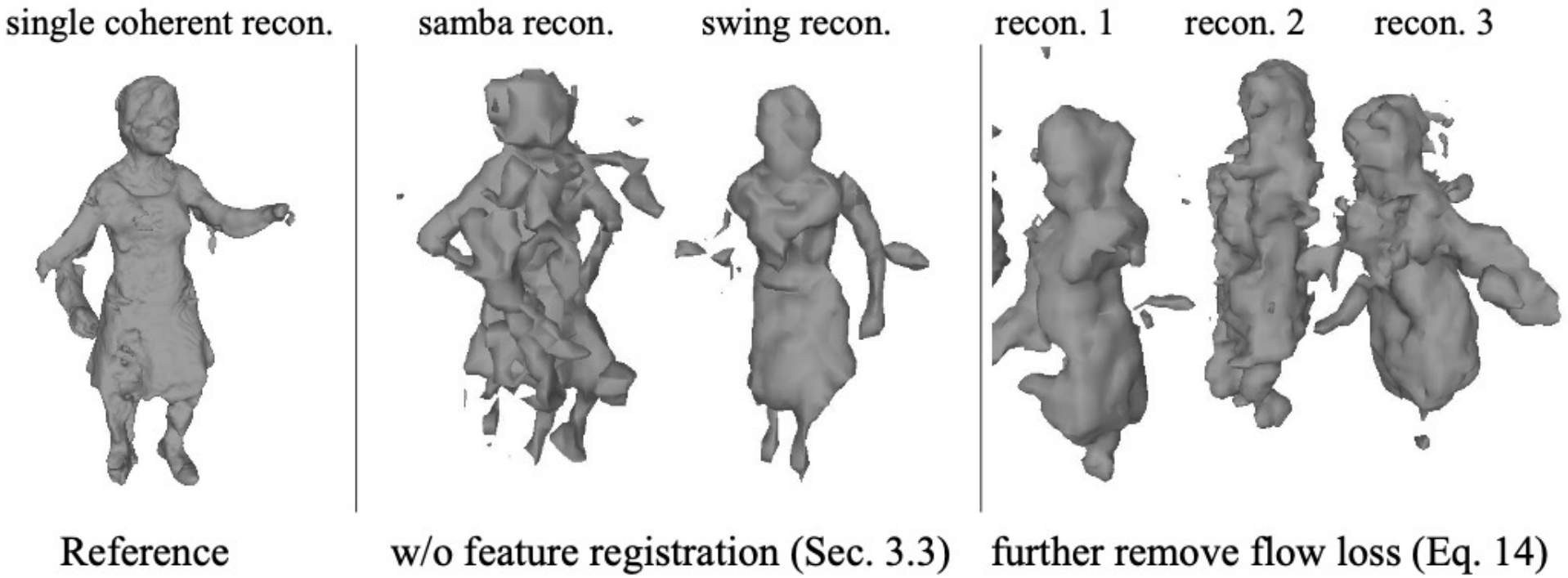


Compliance to topology changes in optimization.

- BANMo incorrectly reconstructs a single rear leg of the dog, but automatically corrects the topology with gradient updates.

Experiments

Diagnostics (Ablation study)



Experiments

Diagnostics (Deformation modeling)

Reference image

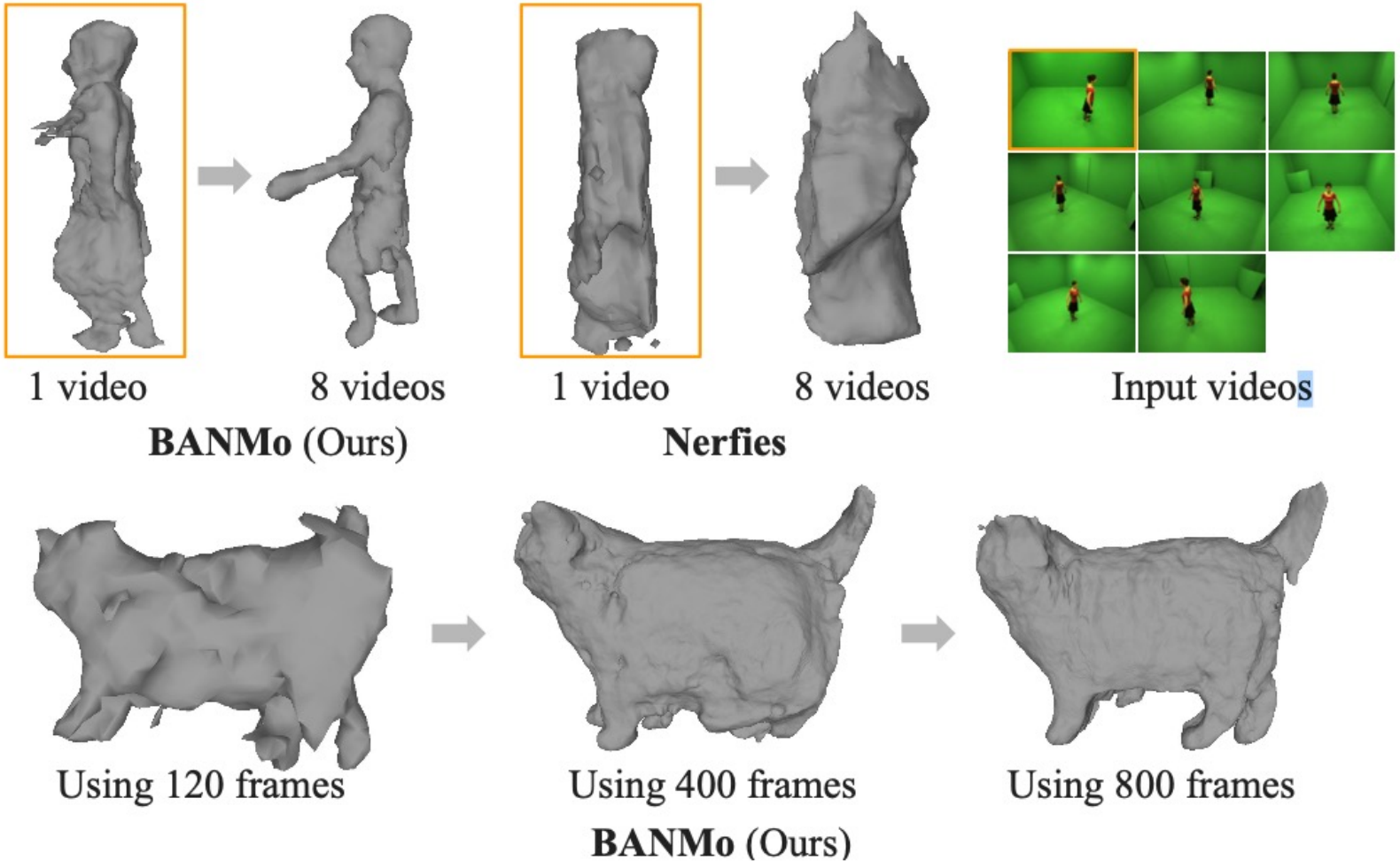
Neural blend skinning (Ours)

MLP-SE(3) (Nerfies)

MLP-translation (NSFF, D-NeRF)

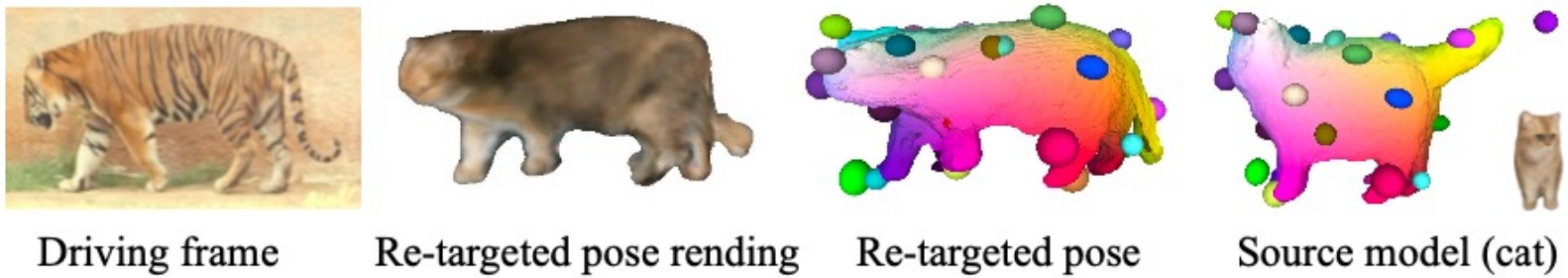
Experiments

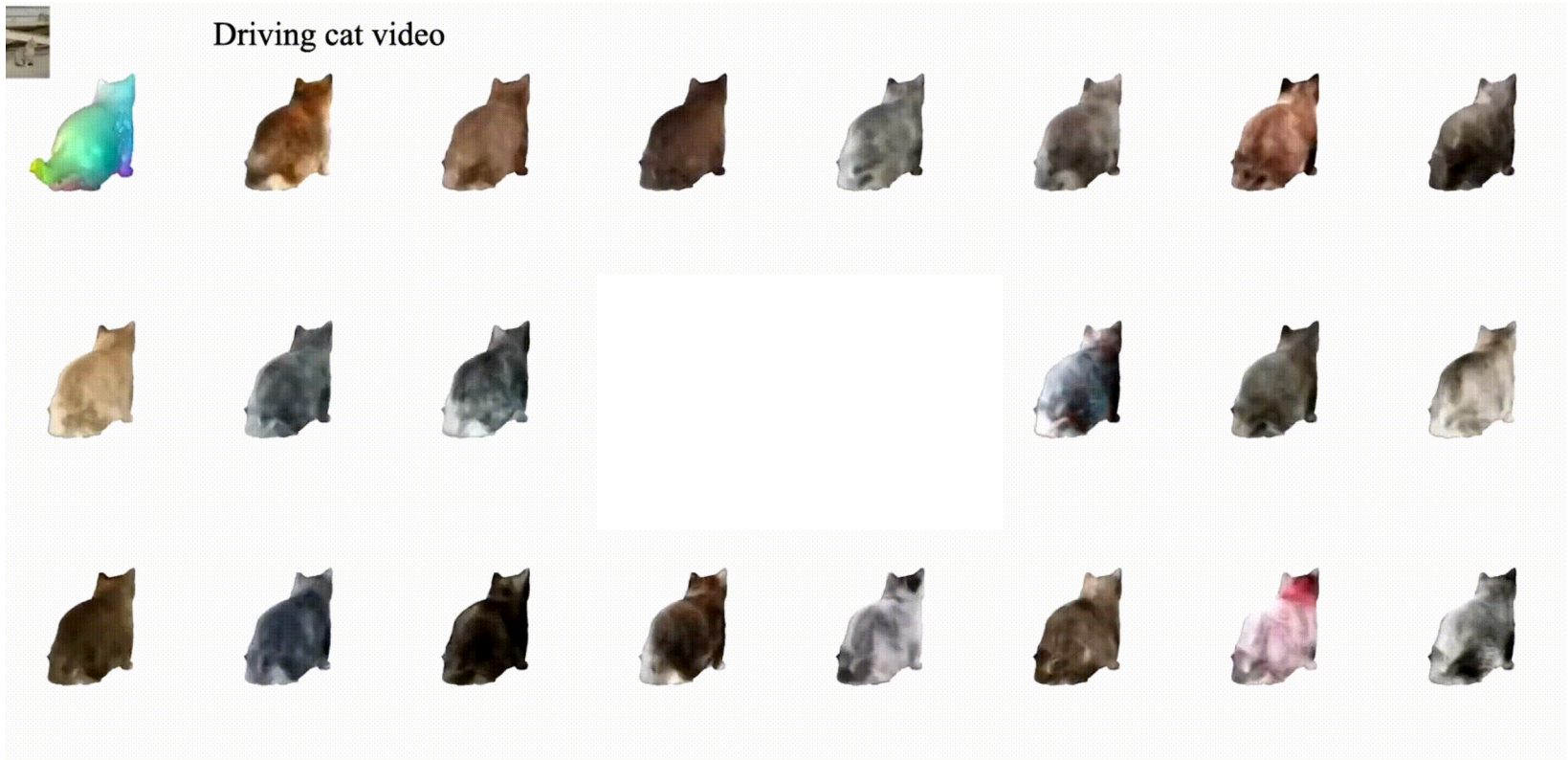
Reconstruction completeness vs number of input videos and video frames



Experiments

Motion retargeting





Conclusion