Do 2D GANs Know 3D Shape? Unsupervised 3D Shape Reconstruction from 2D Image GANs

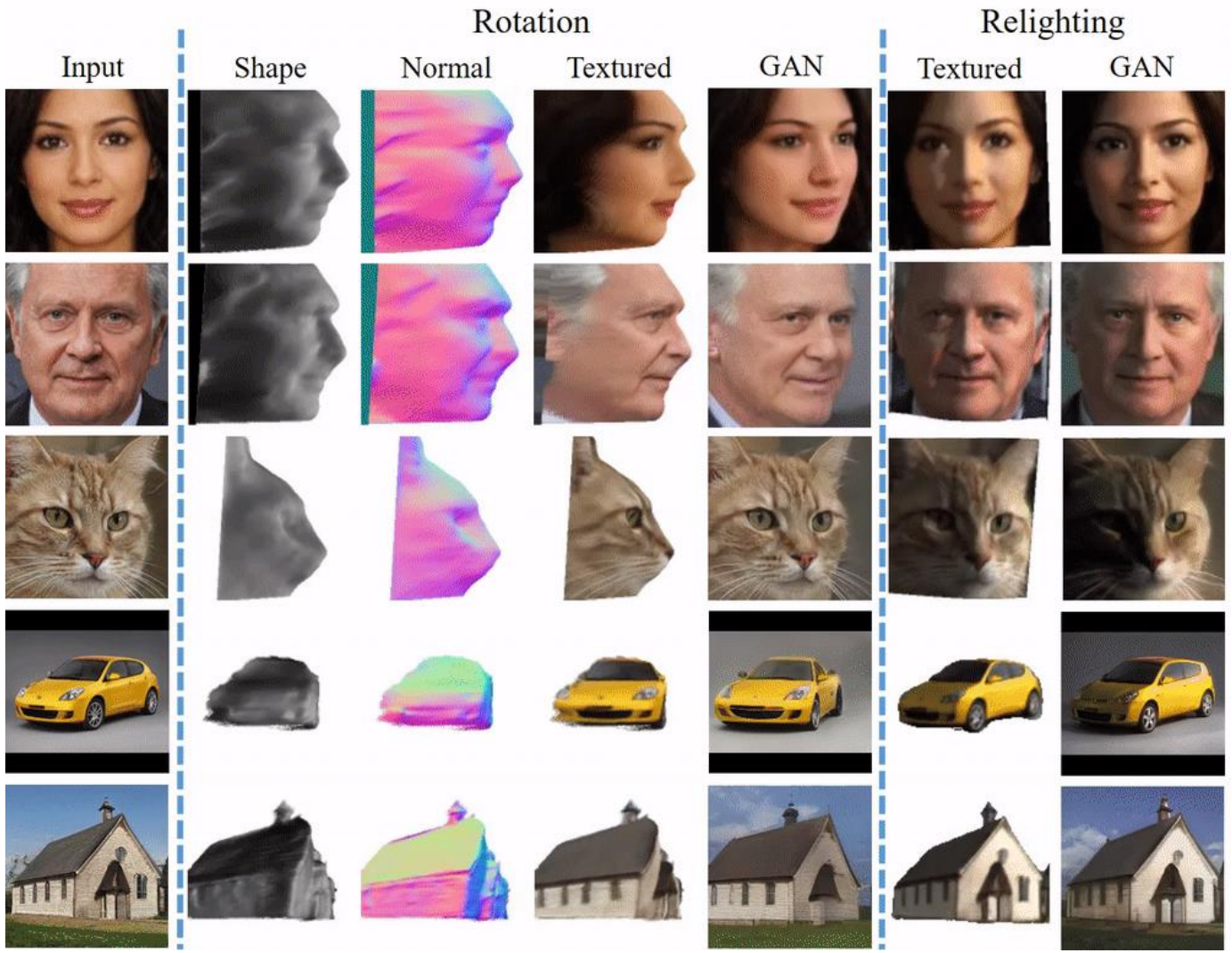
ICLR2021 Oral paper

21.04.12 Leeminsoo

https://arxiv.org/pdf/2011.00844.pdf
https://github.com/XingangPan/GAN2Shape
https://www.youtube.com/watch?v=Ea4eNXf3s24

DAVIAN
Data and Visual Analytics Lab

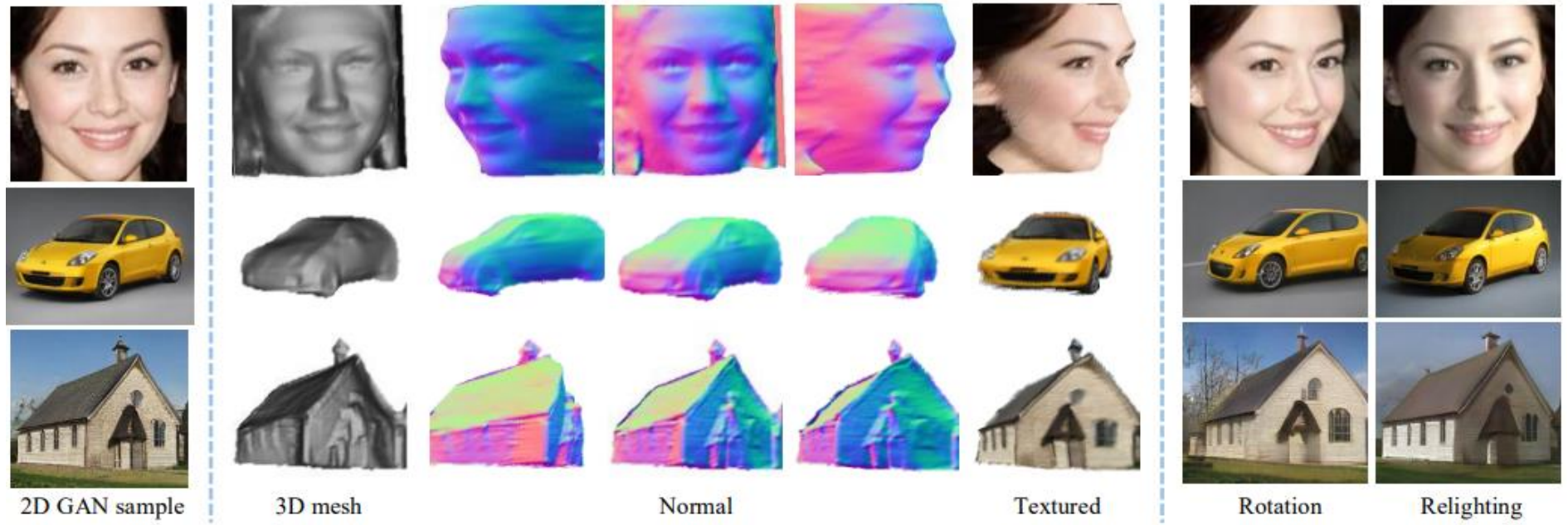
Demo



https://xingangpan.github.io/projects/GAN2Shape.html

Introduction

2D GANs Know 3D Shape? Unsupervised 3D Shape Reconstruction from 2D Image GANs



- The framework is an iterative strategy that explores and exploits diverse viewpoint and lighting variations in the GAN image manifold.

- The framework does not require 2D keypoint or 3D annotations, or strong assumptions on object shapes, yet it successfully recovers 3D shapes.

Background Knowledge

Depth Map

Normal Map

Shading

Albedo

Orginal Img

Other Viewpoint

Background Knowledge

Renderer

$$\hat{\mathbf{I}} = \Phi(\boldsymbol{d}, \boldsymbol{a}, \boldsymbol{v}, \boldsymbol{l})$$

rendered Image

$$= \Pi(\Lambda(\boldsymbol{d}, \boldsymbol{a}, \boldsymbol{l}), \boldsymbol{d}, \boldsymbol{v})$$

reprojection

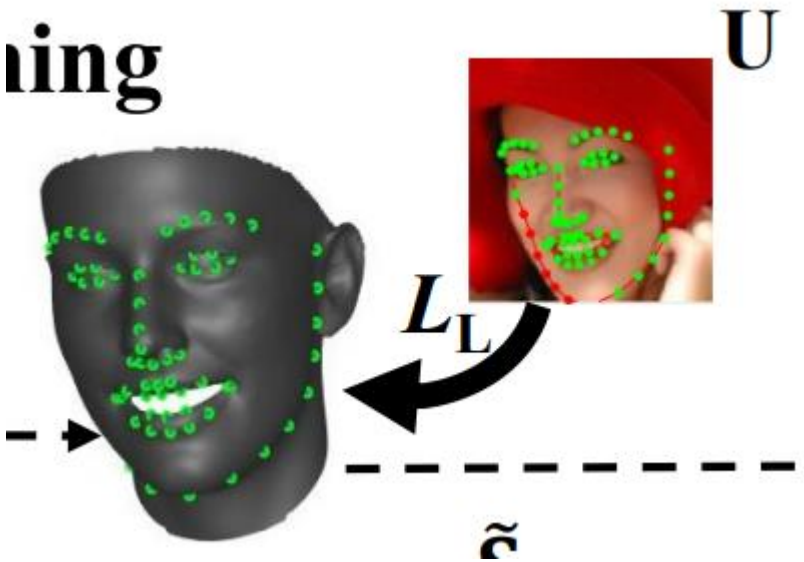
lighting

$$\mathbf{J}_{uv} = (k_s + k_d \max\{0, \langle l, n_{uv} \rangle\}) \cdot a_{uv}$$

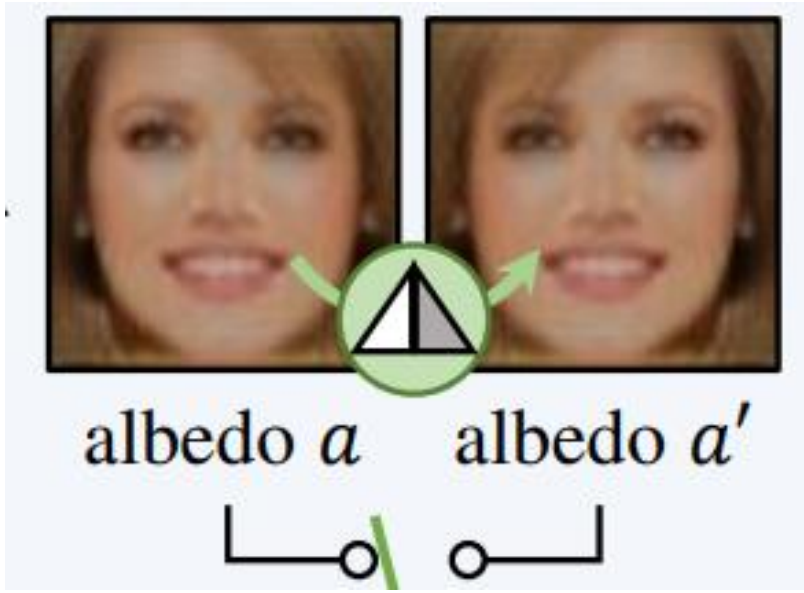
$$\boldsymbol{p} \propto \boldsymbol{K}\boldsymbol{P}, \quad \boldsymbol{K} = \begin{bmatrix} f & 0 & c_i \\ 0 & f & c_j \\ 0 & 0 & 1 \end{bmatrix}, \quad \begin{cases} c_i = \frac{W-1}{2}, \\ c_j = \frac{H-1}{2}, \\ f = \frac{W-1}{2 \tan \frac{\theta_{FOV}}{2}} \end{cases}$$

$$\boldsymbol{p}' \propto \boldsymbol{K}(d_{ij} \cdot \boldsymbol{R}\boldsymbol{K}^{-1}\boldsymbol{p} + \boldsymbol{T})$$

Previous Work



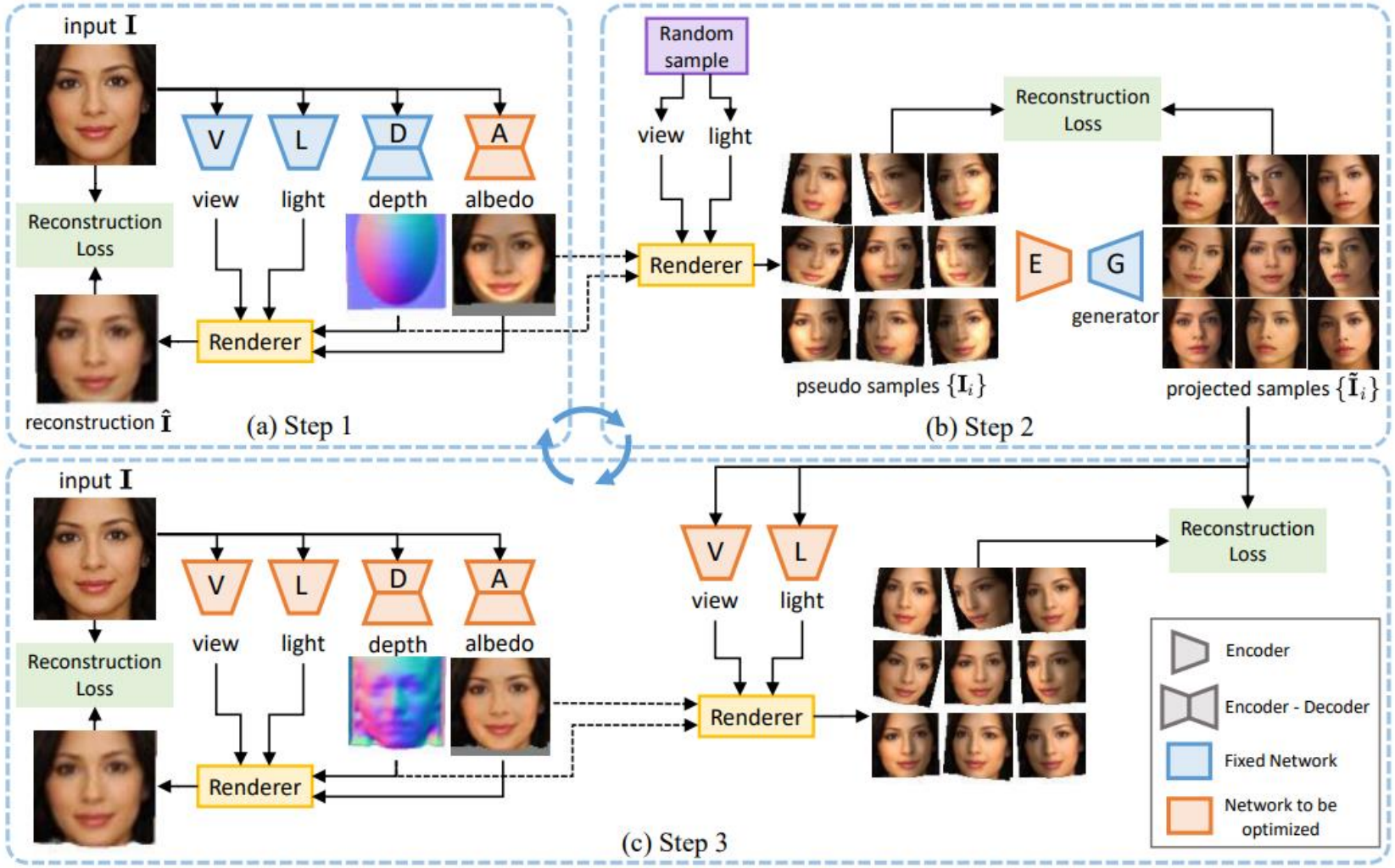
[3D model or 2D keypoints supervision]



[Symmetry Assumption]

Nonlinear 3D Face Morphable Model, CVPR, 2018
Unsupervised Learning of Probably Symmetric Deformable 3D Objects from Images in the Wild, CVPR, 2020

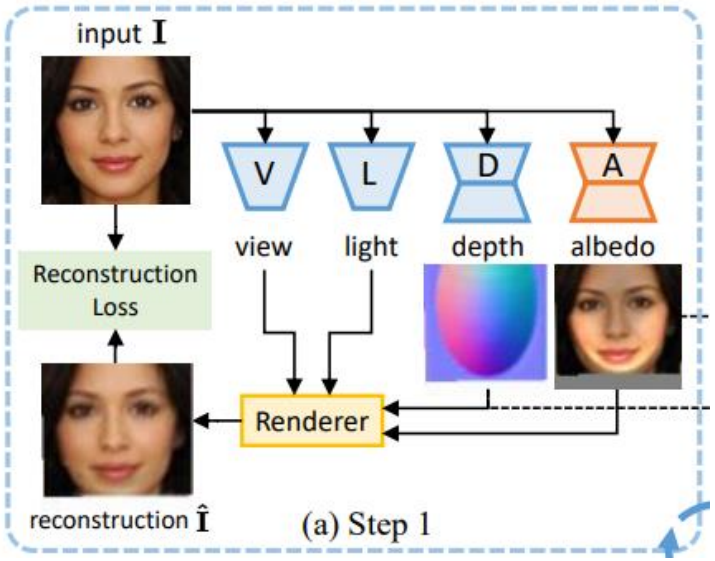
Method



- Creating pseudo samples with different viewpoints and light via GAN to recover 3D shape.

Method

Step 1: Using a Weak Shape Prior.



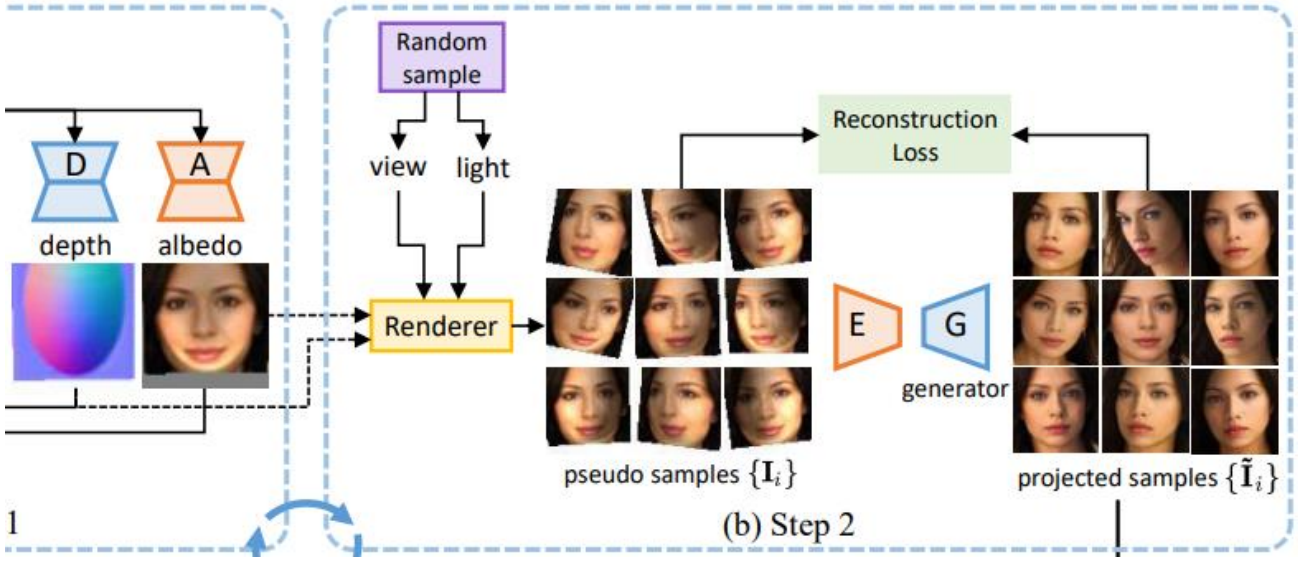
- Initialize
  - view = 0, light = front, depth = ellipsoid
- Renderer

$$\hat{\mathbf{I}} = \Phi(\boldsymbol{d}, \boldsymbol{a}, \boldsymbol{v}, \boldsymbol{l}) = \Pi(\Lambda(\boldsymbol{d}, \boldsymbol{a}, \boldsymbol{l}), \boldsymbol{d}, \boldsymbol{v})$$

- Reconstruction loss
  - Weighted sum of L1 and Perceptual Loss

Method

Step2: Sampling and Projecting to the GAN Image Manifold.



$$\tilde{\mathbf{I}}_i = G(E(\mathbf{I}_i) + \boldsymbol{w})$$

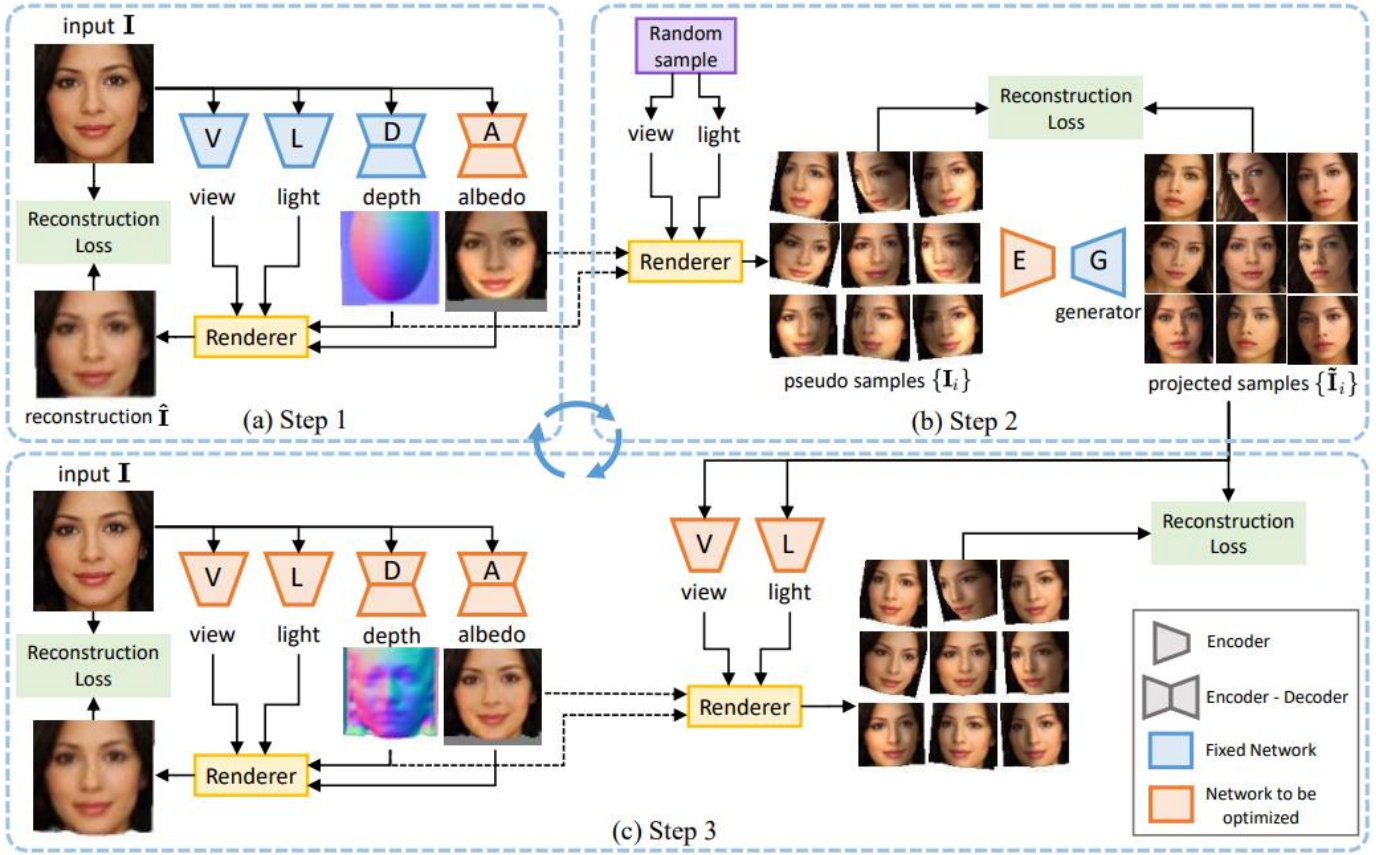
$$\boldsymbol{\theta}_E = \arg\min_{\boldsymbol{\theta}_E} \frac{1}{m} \sum_{i=0}^{m} \mathcal{L}'\left(\mathbf{I}_i, G\big(E(\mathbf{I}_i) + \boldsymbol{w}\big)\right) + \lambda_1 \|E(\mathbf{I}_i)\|_2$$

$$L' = L1 \; of \; the \; discriminator \; features$$

- With the shape prior as an initialization, we are able to create ‘pseudo samples’ by sampling a number of random viewpoints and lighting directions
- In order to leverage such cues to guide novel viewpoint and lighting direction exploration in the GAN image manifold, we perform GAN inversion to these pseudo samples (reconstruct them with the GAN generator G).
- The generator G could regularize the projected samples to lie in the natural image manifold.

Method

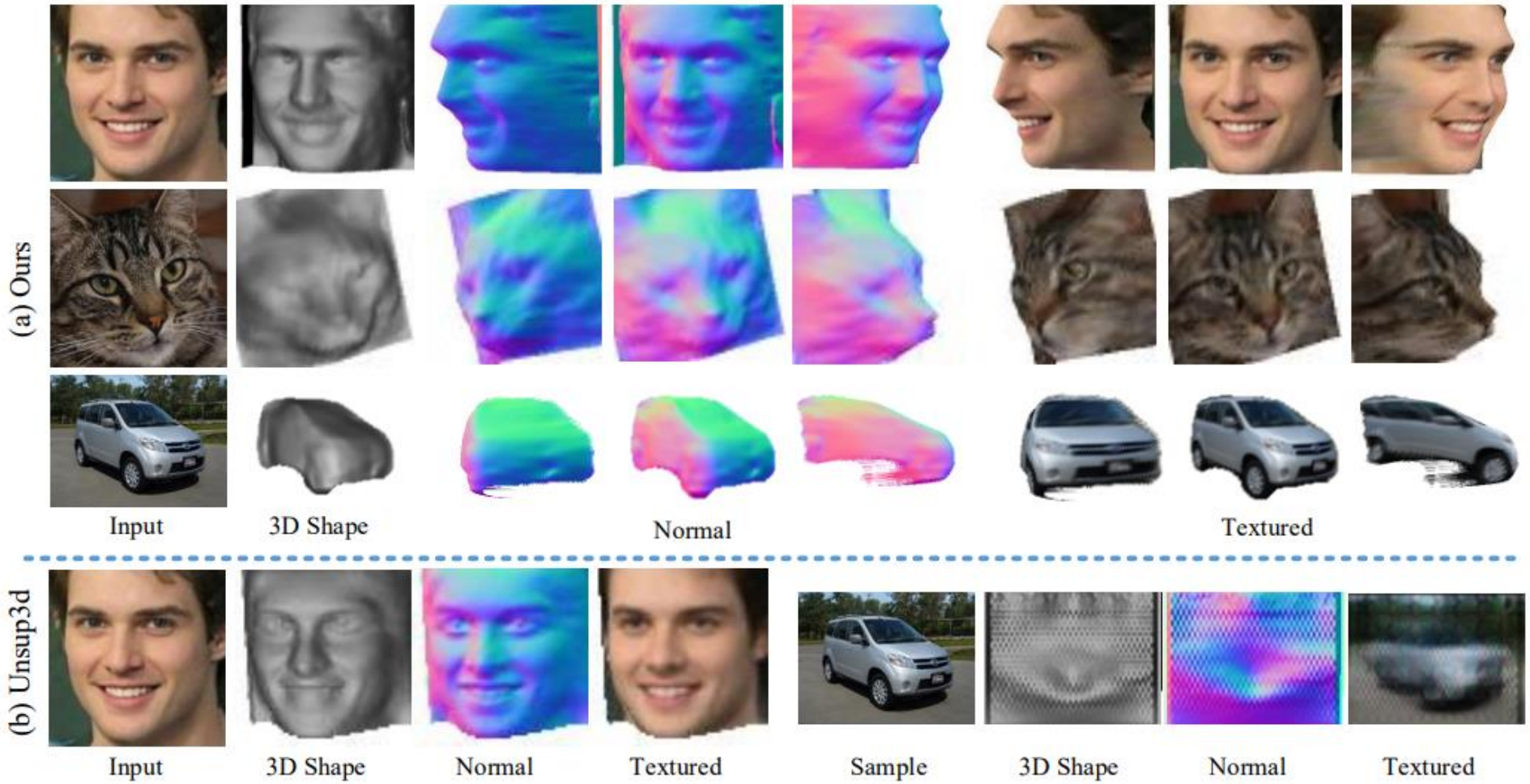
Step3: Learning the 3D Shape.



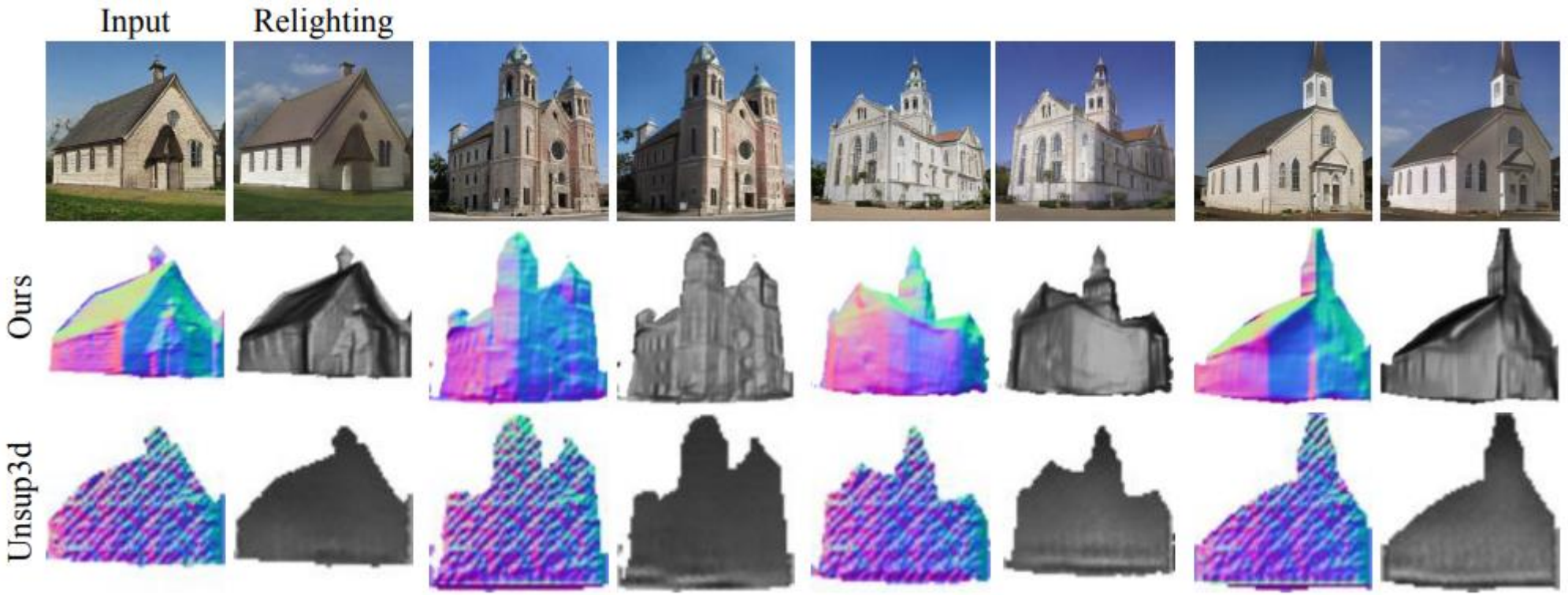
- The projected samples provide images of multiple viewpoint and lighting conditions of nearly the same object content.
- Such single instance multiple view and lighting wetting makes it possible to learn the underlying 3D shape.

$$\boldsymbol{\theta}_D, \boldsymbol{\theta}_A, \boldsymbol{\theta}_V, \boldsymbol{\theta}_L = \underset{\boldsymbol{\theta}_D, \boldsymbol{\theta}_A, \boldsymbol{\theta}_V, \boldsymbol{\theta}_L}{\arg\min} \frac{1}{m} \sum_{i=0}^{m} \mathcal{L}\Big(\tilde{\mathbf{I}}_i, \Phi\big(D(\mathbf{I}), A(\mathbf{I}), V(\tilde{\mathbf{I}}_i), L(\tilde{\mathbf{I}}_i)\big)\Big) + \lambda_2 \mathcal{L}_{smooth}\big(D(\mathbf{I})\big)$$

Experiments



Experiments



Experiments

Table 1: **Comparisons on the BFM dataset.** We report SIDE and MAD errors. 'Symmetry' indicates whether the symmetry assumption on object shape is used. We outperform others on both metrics.

No.	Method	Symmetry	SIDE ($\times 10^{-2}$)↓	MAD (deg.)↓
(1)	Supervised	N	0.419	10.83
(2)	Const. null depth	/	2.723	43.22
(3)	Average g.t. depth	/	1.978	22.99
(4)	Unsup3d (Wu et al., 2020)	Y	0.807	16.34
(5)	Ours (w/o regularize)	Y	0.925	16.42
(6)	Ours	Y	**0.756**	**14.81**
(7)	Unsup3d (Wu et al., 2020)	N	1.334	33.79
(8)	Ours	N	**1.023**	**17.09**

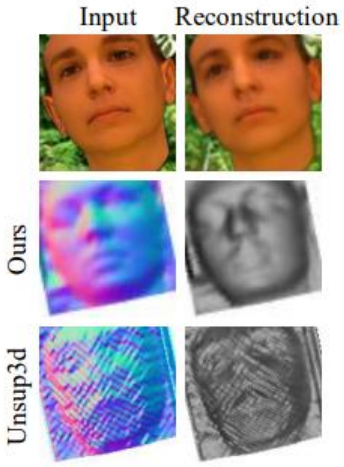
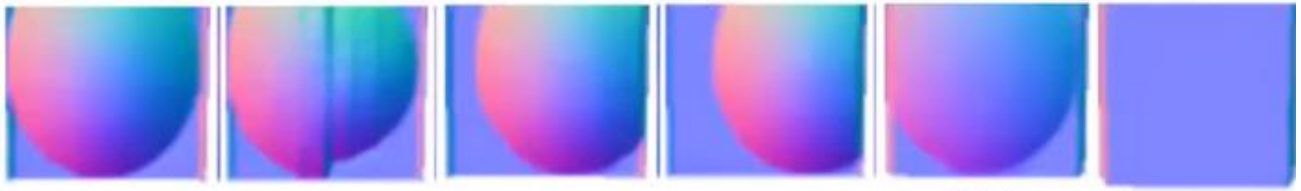


Figure 6: Results without symmetry assumption.

Table 2: **Effects of different shape prior.** We report results of the original ellipsoid shape, asymmetric shape with ellipsoid for the left half and sphere for the right half, shape with its position shifted by 1/6 and 1/4 image width, weaker shape prior whose height is half of the original one, and no shape prior. Qualitative results can be found in Fig.14 in the Appendix.

Shape prior	Origin	Asymmetric	Shift w/6	Shift w/4	Weak	Flat
SIDE ($\times 10^{-2}$)↓	0.756	0.769	0.767	0.775	0.764	1.021
MAD (deg.)↓	14.81	14.95	14.93	15.07	14.97	20.46



Experiments

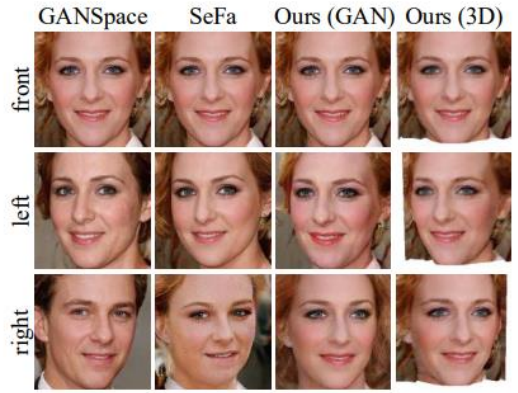


Figure 8: **Qualitative comparison on face rotation.** "Ours (GAN)" and "Ours (3D)" indicate results generated by GAN and rendered from 3D mesh respectively. The face identities in the baseline methods tend to drift during rotation.

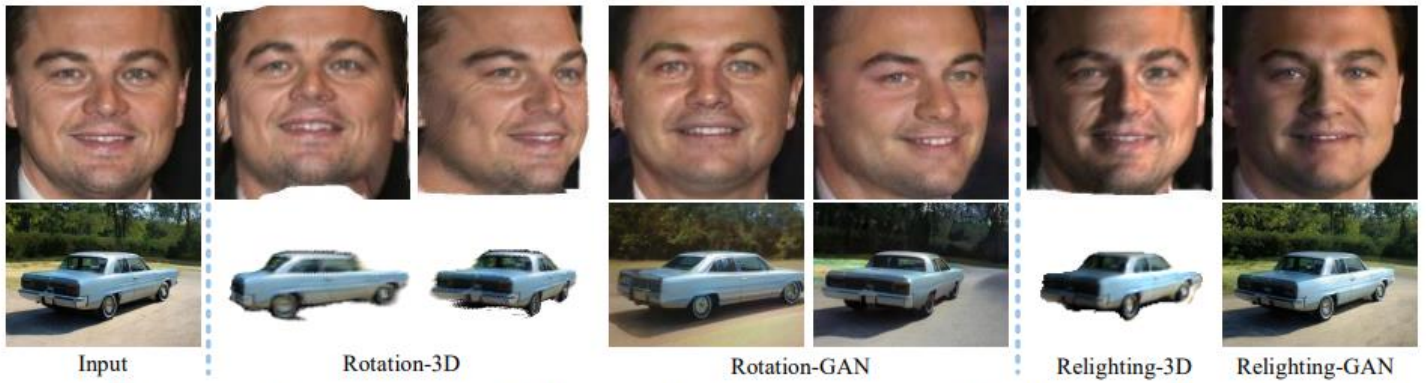


Figure 7: **3D-aware image manipulation**, including rotation and relighting. We show results obtained via both 3D mesh and GANs. The input of the first row is a real natural image. Our method achieves photo-realistic manipulation effects obeying the objects' underlying 3D structures.

Limitation

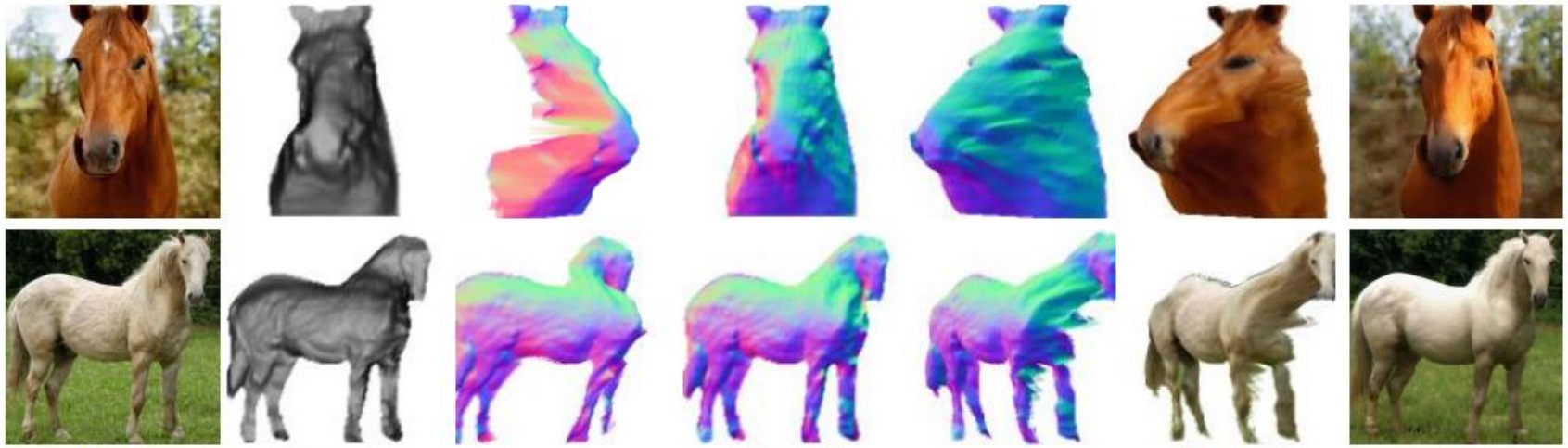


Figure 9: **Qualitative results on the LSUN Horse dataset** (Yu et al., 2015).

- For objects with more sophisticated shapes like horses, a simple convex shape prior may not well reflect the viewpoint and lighting variations.
- The model could not predict the back-side shape of objects.

EOD