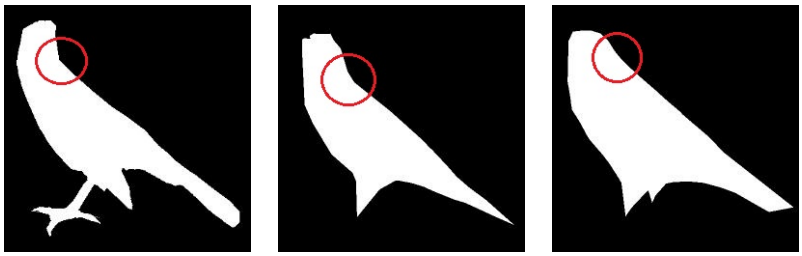
Weakly-supervised Reconstruction of 3D Objects with Large Shape Variation from Single In-the-Wild Images

Shichen Sun[1], Zhengbang Zhu[1], Xiaowei Dai[1], Qijun Zhao[1,2], and Jing Li[1]

[1] College of Computer Science, Sichuan University, China
[2] School of Information Science and Technology, Tibet University, China

In this supplementary material, we first demonstrate the necessity of using structure similarity (SSIM) to evaluate the reconstruction accuracy in Section 1, and then provide in Section 2 the definitions of the edges considered in our model analysis of edge constraints. Section 3 gives the detail of the networks in the paper, and Section 4 presents additional reconstruction results of our method and the state-of-the-art method CMR [1] on CUB-200-2011.



(a) Ground Truth (b) Ours (c) CMR

Fig. 1: Example for showing the necessity of SSIM as performance metric to assess the reconstruction accuracy. Although our reconstructed shape is visually more consistent with the ground truth, our method has higher SSIM but lower IoU than the counterpart method CMR (see Table 1).

1 SSIM as Performance Metric

Here, we use an example to show the necessity of using structure similarity (SSIM) as performance metric to assess the reconstruction accuracy. See Fig. 1. Obviously, compared with the result of the CMR method [1], our reconstructed shape is visually more consistent with the ground truth, which is correctly measured by SSIM. However, as shown in Table 1, CMR has higher IoU than our method for this example. We believe that this is because IoU puts more weight on the interior of reconstructed objects, while neglecting to some extent the

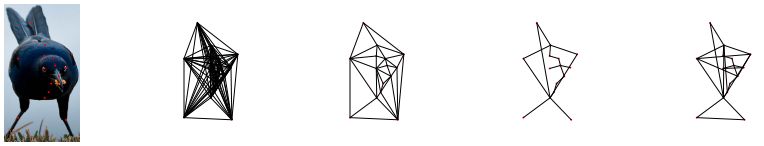
discrepancy of boundary. Therefore, in this paper we propose to use SSIM as another performance metric to measure the similarity between the reconstructed and ground truth shapes.

Table 1: The reconstruction accuracy of our method and the counterpart method CMR for the example image in Fig. 1 in terms of IoU and SSIM.

Method	IoU	SSIM
Ours	0.681	**0.81635**
CMR	**0.718**	0.80799

2 Edge Definitions

To more effectively utilize the geometric constraints in training data and to compensate for the shortcoming of mean squared error (MSE) in describing shape topology, our proposed method defines inter-keypoint constraint based on the edges connecting the keypoints. In our experiments, we implement four different definitions of the edges. (i) FC: edges collecting all pairs of visible keypoints; (ii) DT: edges defined by Delaunay Triangulation over the set of visible keypoints; (iii) M1: a set of edges manually defined according to prior knowledge; and (iv) M2: another set of edges also manually defined according to prior knowledge. Figure 2 shows the edges for an example bird image. As being demonstrated by our experiments, our proposed DT edges achieve the best performance thanks to its capability of dealing with large pose variations and to its high efficiency.



(a) RGB image (b) fc (c) DT (d) DT (e) DT

Fig. 2: Visualization of the four different definitions of edges in implementing the inter-keypoint constraint.

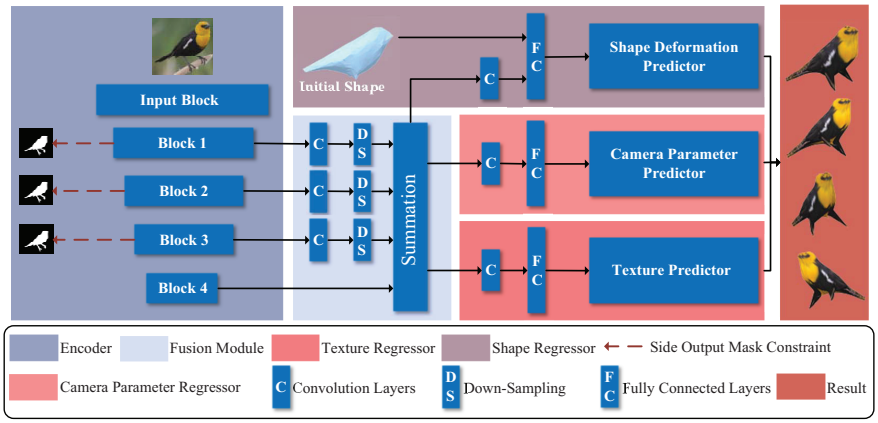


Fig. 3: Schematic diagram of the proposed method of unsupervised 3D object reconstruction from single images.

3 Network Details

3.1 Proposed Network

Figure 3 shows the architecture of the proposed network. We take the pre-trained ResNet-18 [2] as the encoder to extract features from input RGB images. As shown in Fig. 4, features extracted by different blocks in the endcoder are sent to the fusion module. Shape deformation, UV-flow [1] and camera parameters [3] are then predicted all by inference from the fused features but with respective regressors. In each regressor, the fused feature is first transformed via the feature transformation module (see Fig. 5) and then sent to the corresponding predictor. Figures 6 and 7 show the architecture of the three predictors. The layers involved in Fig. 5 - Fig. 7 are defined as follows:

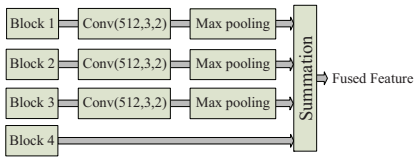


Fig. 4: Architecture of the fusion module. The input of the fusion module is the feature maps extracted by the four blocks in the encoder.

- Linear(a) is an affine transformation layer [4]. a is the number of feature maps.
- Conv(a, b, c) is a 2D convolution layer. The number of feature maps is a, the kernel size is $b \times b$, and the stride size is $c \times c$.
- Upsampling(a) uses bilinear interpolation to upsample the feature maps. a is the scale fator.
- LeakyReLU(a) is a nonlinear activation function [5]. a is the value of alpha.

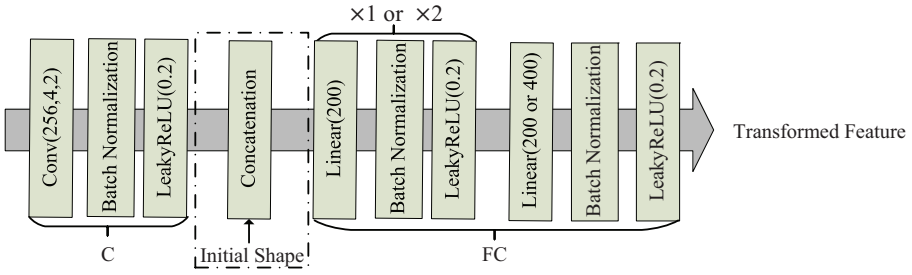


Fig. 5: Architecture of the feature transformation module. The number of feature maps of the last linear transformation layer in shape deformation regressor is 400. The concatenation with initial shape is applied only for the shape deformation regressor.

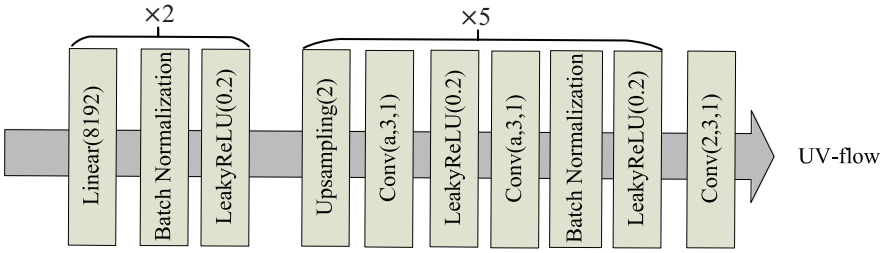
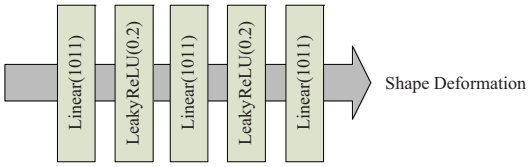
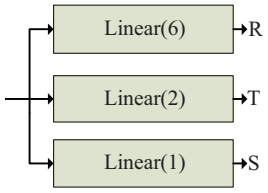


Fig. 6: Architecture of the texture predictor. The size of UV-flow is $128 \times 64 \times 2$. $a = \{128, 64, 32, 16, 8\}$.



(a) Shape deformation predictor.



(b) Camera parameters predictor. The three linear transformation layers estimate the Roation (R), Transformation (T), Scale (S) parameters, respectively.

Fig. 7: Architecture of (a) shape deformation predcitor and (b) camera parameters predictor.

4 Additional Reconstruction Results

Figures 8 and 9 visualize some additional results by our method and the counterpart method [1] for 3D bird reconstruction.

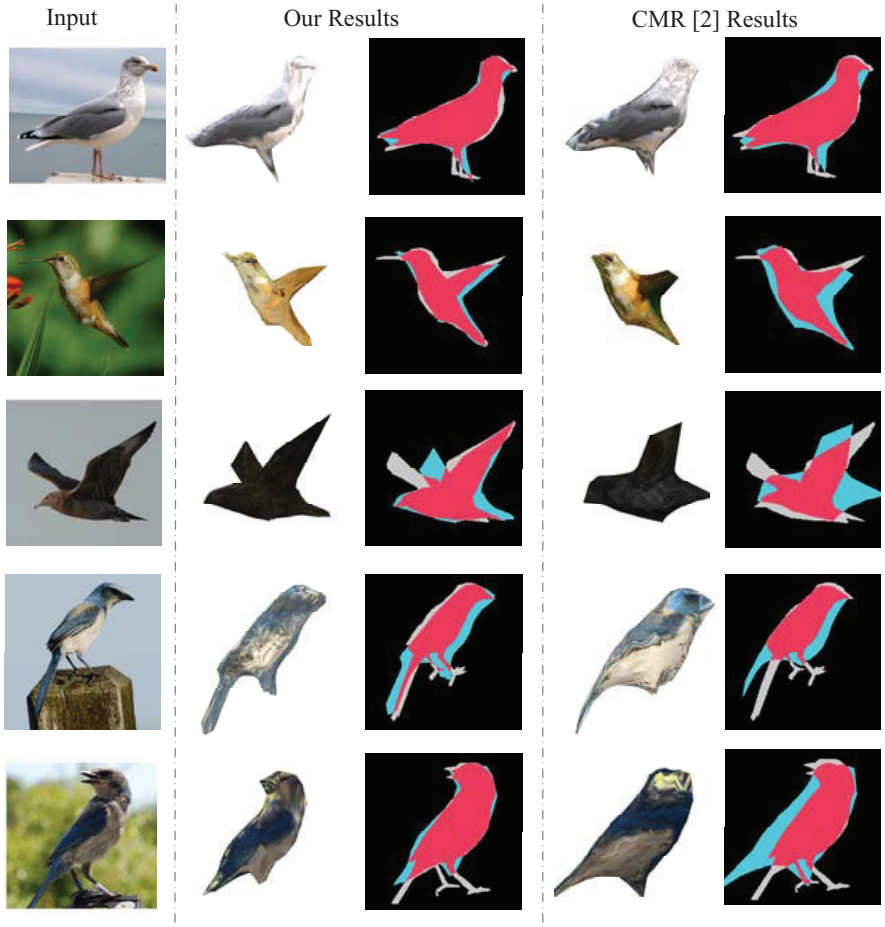


Fig. 8: Reconstruction results of our method and the CMR method [1] on CUB-200-2011. For easy comparison, we overlay ground truth masks (in grey color) with the reconstructed ones (in cyan color), and highlight their common regions in red color.

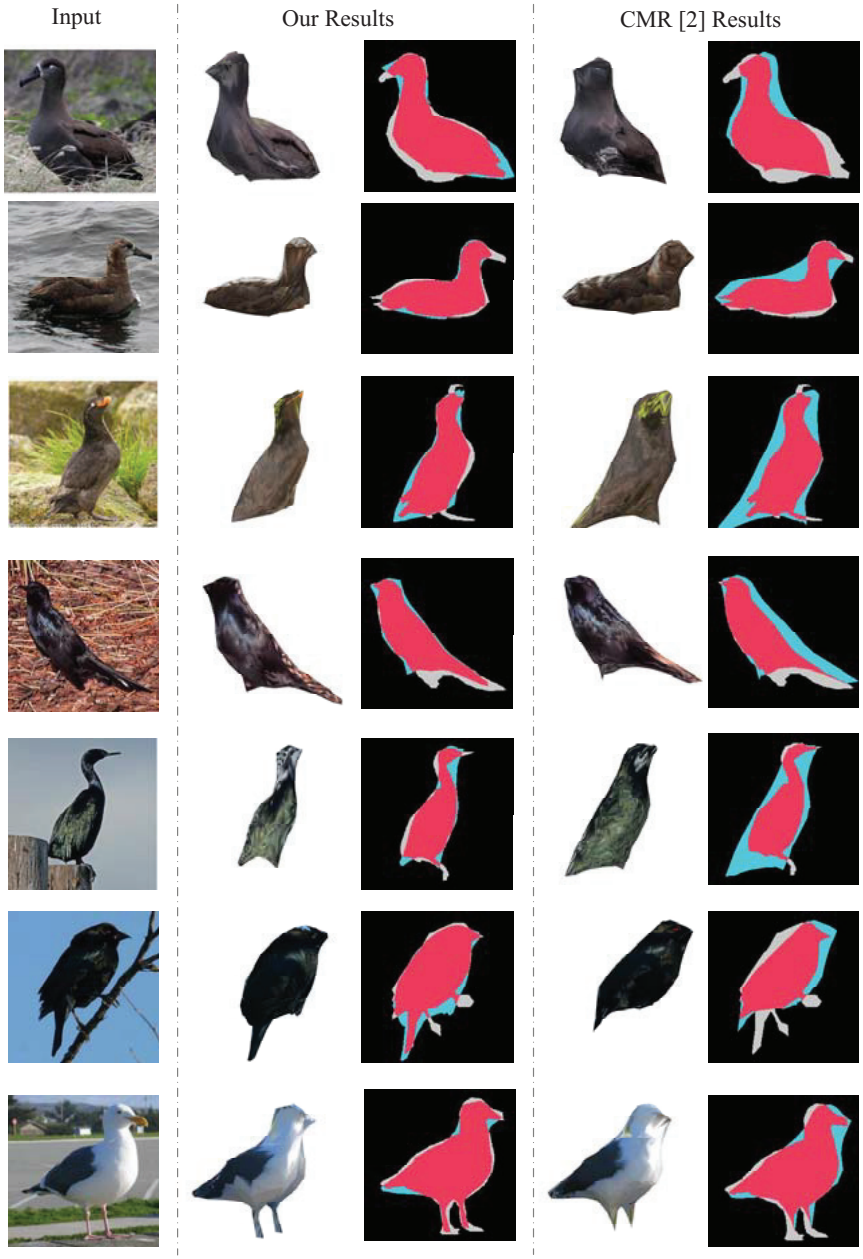


Fig. 9: Reconstruction results of our method and the CMR method [1] on CUB-200-2011. For easy comparison, we overlay ground truth masks (in grey color) with the reconstructed ones (in cyan color), and highlight their common regions in red color.

References

1. Kanazawa, A., Tulsiani, S., Efros, A.A., Malik, J.: Learning category-specific mesh reconstruction from image collections. In: European Conference on Computer Vision (ECCV). (2018) 371–386
2. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2016) 770–778
3. Zhou, Y., Barnes, C., Jingwan, L., Jimei, Y., Hao, L.: On the continuity of rotation representations in neural networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2019) 5745–5753
4. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. In: Proceedings of the IEEE. (1998) 2278–2324
5. Maas, A.L., Hannun, A.Y., Ng, A.Y.: Rectifier nonlinearities improve neural network acoustic models. In: International Conference on Machine Learning (ICML) Workshop on Deep Learning for Audio, Speech and Language Processing. (2013) 3