VERTEX: VEhicle Reconstruction and TEXture Estimation from a Single Image Using Deep Implicit Semantic Template Mapping

Xiaochen Zhao[1], Zerong Zheng[1], Chaonan Ji[1], Zhenyi Liu[2], Siyou Lin[1], Tao Yu[1], Jinli Suo[1], and Yebin Liu[1(✉)]

[1] Tsinghua University, Beijing 100084, China
liuyebin@mail.tsinghua.edu.cn
[2] Jilin University, Changchun 130015, China

Abstract. We introduce VERTEX, an effective solution to recovering the 3D shape and texture of vehicles from uncalibrated monocular inputs under real-world street environments. To fully utilize the semantic prior of vehicles, we propose a novel geometry and texture joint representation based on implicit semantic template mapping. Compared to existing representations which infer 3D texture fields, our method explicitly constrains the texture distribution on the 2D surface of the template and avoids the limitation of fixed topology. Moreover, we propose a joint training strategy that leverages the texture distribution to learn a semantic-preserving mapping from vehicle instances to the canonical template. We also contribute a new synthetic dataset containing 830 elaborately textured car models labeled with key points and rendered using Physically Based Rendering (PBRT) system with measured HDRI skymaps to obtain highly realistic images. Experiments demonstrate the superior performance of our approach on both testing dataset and in-the-wild images. Furthermore, the presented technique enables additional applications such as 3D vehicle texture transfer and material identification, and can be generalized to other shape categories.

Keywords: Vehicle 3d reconstruction · Implicit representation

1 Introduction

Monocular visual scene understanding is a fundamental technology for many automatic applications, especially in the field of autonomous driving. Using only a single-view driving image, available vehicle parsing studies have covered popular topics starting from 2D vehicle detection, then 6D vehicle pose recovery,

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-20497-5_52.

L. Fang et al. (Eds.): CICAI 2022, LNAI 13604, pp. 634–646, 2022.
https://doi.org/10.1007/978-3-031-20497-5_52

and finally vehicle shape reconstruction. However, much less efforts are devoted to vehicle texture estimation, even though both humans and autonomous cars heavily rely on the appearance of vehicles to perceive surroundings. Simultaneously recovering the geometry and texture of vehicles is also important for synthetic driving data generation [19], vehicle tracking [20], vehicle parsing [23] and so on.

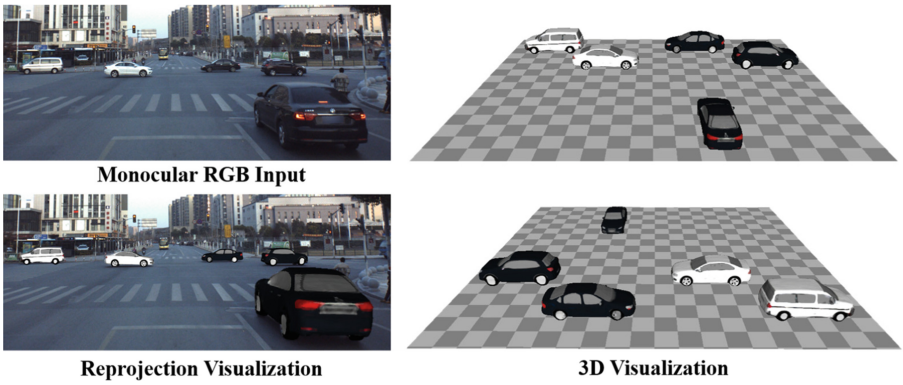

Fig. 1. We propose a method to recover realistic 3D textured models of vehicles from a single image (top left) under real street environments. Our approach can reconstruct the shape and texture with fine details. (We manually adjust the scale and layout of models for better visualization.)

Challenges for monocular geometry and texture recovery of vehicles mainly arise from the difficulties in inferring the invisible texture conditioned on only visible pixels while handling various vehicle shapes. Additionally, in real-world street environments, reconstruction methods are also expected to offset the adverse impact of complicated lighting conditions (e.g., strong sunlight and shadows) and diverse materials (e.g., transparent or reflective non-Lambertian surfaces). That said, the shape and appearance of vehicles are not completely arbitrary. Our key insight is that those challenges can be addressed with the prior knowledge from vehicle models, especially the part semantics. Therefore, we seek to find a method that is a) aware of the underlying semantics of vehicles, and b) flexible enough to recover various geometric structures and texture patterns.

Recently, deep implicit functions (DIFs), which model 3D shapes using continuous functions in 3D space, have been proven powerful in representing complex geometric structures [22,28]. Texture fields (TF) [26] and PIFu [31] took a step further by representing mesh texture with implicit functions and estimating point color conditioned on the input image. To do so, both TF and PIFu diffuse the surface color into the 3D space. However, it remains physically unclear how to define and interpret the color value off the surface. What's worse, geometry and texture are not fully disentangled in either PIFu or TF, as they rely on the

location of surface to diffuse the color into the 3D space, making it difficult to incorporate semantic constraints.

In this paper, we explore a novel method, VERTEX, for VEhicle Reconstruction and TEXture estimation from a single image in real-world street environments. At its core is a novel implicit geo-tex representation that extends DIFs and jointly represents vehicle surface geometry and texture using implicit semantic template mapping. The key idea is to map each vehicle instance to a canonical template field [8,39] in a semantic-preserving manner. In our geo-tex representation, texture inference is constrained on the 2-manifold of the canonical template; in this way, we can leverage the semantic prior of vehicle template, encourage the model to learn a consistent latent space for all vehicles and bypass the unclear physical meaning of a texture field.

However, training such a representation for vehicle reconstruction is not straight-forward, because we have no access to the ground-truth mapping from vehicle instances to the canonical template field. [8,39] proposed to train the mapping network in an unsupervised manner, and the mapping follows the principle of shortest distance. As a result, the mapping in these methods is not guaranteed to preserve accurate semantic correspondences. To resolve this drawback, we propose a joint training method for the geometry reconstruction and texture estimation networks. Our training method is largely different from the training schedule of "first geometry then texture" adopted by typical reconstruction works [13,26,31]. This stems from the insight that the surface texture is closely related to its semantic labels; consider the appearance difference between different parts such as car bodies, windows, tires and lights as examples. The texture information can serve as the additional supervision to force the template mapping to be semantic-preserving.

Trained with our joint training method, our implicit geo-tex representation owns the advantages of both mesh templates and implicit functions: on one hand, it is expressive to represent various shapes, which is the main advantage of DIFs; on the other hand, it disentangles texture representation from geometry, thus supports many downstream tasks including material editing and texture transfer. Although it is initially designed for vehicles, our method can generalize to other objects such as bikes, planes and sofas.

To simulate real street environments and evaluate our method, we also contribute a synthetic dataset containing 830 elaborately textured car models rendered using Physically Based Rendering (PBRT) system with measured HDRI skymaps to obtain highly realistic images. Each instance is labeled with key points as semantic annotations and can be exploited for evaluation and future research.

In summary, our contributions include:

- a novel implicit geo-tex representation with semantic dense correspondences and latent space disentanglement, enabling fine-grained texture estimation, part-level understanding and vehicle editing;
- a joint training strategy leveraging the consistency between RGB color and part semantics for semantics-preserving template mapping;

– a new vehicle dataset, containing diverse detailed car CAD models, PBRT based rendered images and corresponding real-world HDRI skymaps.

2 Related Work

2.1 Monocular Vehicle Reconstruction

Recently, many works [1,10,13,16] concentrate on vehicle 3D texture recovery under real environments. Due to the lack of ground truth 3D data of real scenes, they mainly focus on the reconstruction from collections of 2D images utilizing unsupervised or self-supervised learning and build on mesh representation. Though eliminating the need for 3D annotations and generating meaningful vehicle textured models, these works still suffer from coarse reconstruction results and the limitation of fixed-topology representation. With large-scale synthetic datasets such as ShapeNet [4], many works [6,26,33] train deep neural networks to perform vehicle reconstruction from images. Based on volumentrically representation like 3D voxel [33] and implicit functions [26], these works generate plausible textured models in the synthetic dataset, but still struggle with low-quality texture. In contrast, our approach outperforms state-of-the-art methods in terms of visual fidelity and 3D consistency while representing topology-varying objects.

In addition, some works [3,9,25,27,38,40] focus on novel view synthesis, i.e., inferring texture in 2D domain. Although they can produce realistic images, they lack compact 3D representation, which is not in line with our goal.

2.2 Deep Implicit Representation

Traditionally, implicit functions represent shapes by constructing a continuous volumetric field and embed meshes as its iso-surface [2,32,34]. In recent years, implicit functions have been implemented with neural networks [5,11,22,28,31,37] and have shown promising results. For example, DeepSDF [28] proposed to learn an implicit function where the network output represents the signed distance of the point to its nearest surface. Other approaches define the implicit functions as 3D occupancy probability functions and cast shape representation as a point classification problem [5,22,31,37].

As for texture inference, both TF [26] and PIFu [31] define texture implicitly as a function of 3D positions. The former uses global latent codes separately extracted from input the image and geometry whereas the latter leverages local pixel-aligned features. Compared with the above approaches [26,31] which predict texture distribution in the whole 3D space, our method explicitly constrains the texture distribution on the 2D manifold of the template surface with implicit semantic template mapping.

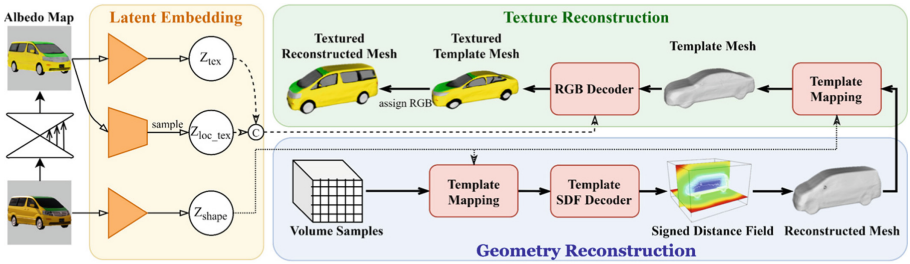

Fig. 2. The overview of our approach. Given the single RGB image, vehicle digitization is achieved by geometry and texture reconstruction. We first convert the original picture into an albedo map, and then extract multi-scale latent codes in Latent Embedding. Conditioned on these latent codes, our neural networks can infer SDF to reconstruct mesh surface and then regress RGB value for the surface.

3 Implicit Geo-Tex Representation

Our method for vehicle reconstruction and texture estimation is built upon a novel geo-tex joint representation, which is presented in this section.

3.1 Basic Formulation

We believe that an ideal geo-tex representation should disentangle texture representation from geometry as uv mapping does and should be accord with the physical fact that texture only attaches to the 2D surface of the object. In particular, observing that vehicles are a class of objects with a strong template prior, we extend DIT [39] and propose a *joint* geo-tex representation using deep implicit semantic templates. The key idea is to manipulate the implicit field of the vehicle template to represent vehicle geometry while embedding texture on the 2-manifold of the template surface. Mathematically, we denote the vehicle template surface with $\mathcal{S}_T$ as the level set of a signed distance function $F : \mathbb{R}^3 \mapsto \mathbb{R}$, i.e. $F(\boldsymbol{q}) = 0$, where $\boldsymbol{q} \in \mathbb{R}^3$ denotes a 3D point. Then our representation can be formulated as:

$$\begin{cases} \boldsymbol{p}_{tp} = W(\boldsymbol{p}, \boldsymbol{z}_{shape}) \\ s = F(\boldsymbol{p}_{tp}) \\ \boldsymbol{p}_{tp}^{(S)} = W(\boldsymbol{p}^{(S)}, \boldsymbol{z}_{shape}) \\ c = T(\boldsymbol{p}_{tp}^{(S)}, \boldsymbol{z}_{tex}) \end{cases} \tag{1}$$

where $W : \mathbb{R}^3 \times \mathcal{X}_{shape} \mapsto \mathbb{R}^3$ is a spatial warping function mapping the 3D point $\boldsymbol{p} \in \mathbb{R}^3$ to the corresponding location $\boldsymbol{p}_{tp}$ in the canonical template space conditioned on the shape latent code $\boldsymbol{z}_{shape}$, and F queries the signed distance value s at p_{tp}. $\boldsymbol{p}^{(S)} \in \mathcal{S} \subset \mathbb{R}^3$ is a 3D point on the vehicle surface $\mathcal{S}$, which is also mapped onto the template surface $\mathcal{S}_T$ using the warping function W, and $T : \mathcal{S}_T \times \mathcal{X}_{tex} \mapsto \mathbb{R}^3$ regresses the color value c of the template surface point $\boldsymbol{p}_{tp}^{(S)}$ conditioned on the texture latent code $\boldsymbol{z}_{tex}$. Intuitively, we map the vehicle

surface to the template using warping function W and embed the surface texture of different vehicles onto one unified template. Therefore, in our representation, texture is only defined on the template surface (a 2D manifold), avoiding unclear physical meaning of a three-dimensional texture field.

3.2 Formulation for Image-based Reconstruction

For a specific instance, the shape information is defined by $\boldsymbol{z}_{shape}$, while the texture information is encoded as $\boldsymbol{z}_{tex}$, both of which can be extracted from the input image using CNN-based encoders. To further preserve fine details presented in the monocular observation, we fuse local texture information represented as $\boldsymbol{z}_{loc\_tex}(\boldsymbol{p})$ at the pixel level. Not only the texture in visible region can benefit from local features, invisible regions can also be enhanced with the structure prior of the template. Formally, our formulation can be rewritten as:

$$\begin{cases} \boldsymbol{p}_{tp} = \mathcal{W}(\boldsymbol{p}, \boldsymbol{z}_{shape}) \\ s = F(\boldsymbol{p}_{tp}) \\ \boldsymbol{p}_{tp}^{(S)} = W(\boldsymbol{p}^{(S)}, \boldsymbol{z}_{shape}) \\ c = T(\boldsymbol{p}_{tp}^{(S)}, \boldsymbol{z}_{tex}, \boldsymbol{z}_{loc\_tex}(\boldsymbol{p})) \end{cases} \tag{2}$$

where $T : \mathcal{S}_T \times \mathcal{X}_{tex} \times \mathcal{X}_{loc\_tex} \mapsto \mathbb{R}^3$ is conditioned on the latent codes $\boldsymbol{z}_{tex} and \boldsymbol{z}_{loc\_tex}$.

In summary, aiming at vehicle texture recovery, our representation is more expressive with less complexity. However, implementing and training our representation for textured vehicle reconstruction is not straight-forward. We will introduce how we achieve this goal in Sect. 4.

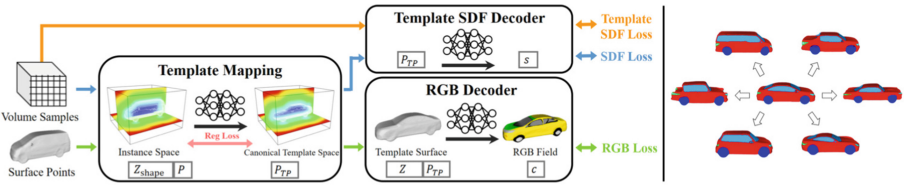

Fig. 3. To implement implicit semantic template mapping (right), we minimize both data terms of geometry (blue arrows) and texture (green arrows) reconstruction simultaneously. Besides, the regularization terms (orange and pink arrows) for specific network modules are applied to assist training. Note that Z in RGB Decoder is the concatenation of the global and local texture latent codes. (Color figure online)

4 Joint Geo-tex Training Method

4.1 Network Architecture

Figure 2 illustrates the overview of our network, consisting of three modules, i.e., Latent Embedding (yellow), Geometry Reconstruction (blue) and Texture

Estimation (green). Our network takes as input a single vehicle image and corresponding 2D silhouette, which can be produced by off-the-shelf 2D detectors [15], and generates a textured mesh.

Albedo Recovery: We empirically found that directly extracting texture latent codes from the input images leads to unsatisfactory results. Therefore, before feeding the input image to our network, we first infer the intrinsic color in 2D domain by means of image-to-image translation [30], and the recovered albedo image will be used as the input for texture encoders in Latent Embedding. We find this module effectively contributes to alleviating the noise effects of image illumination on consistent texture recovery.

Latent Embedding: The global shape and texture latent codes, $\boldsymbol{z}_{shape}$ & $\boldsymbol{z}_{tex}$, are extracted from the input image and recovered albedo map using two separate ResNet-based [12] encoders respectively. The local texture feature, $\boldsymbol{z}_{loc\_tex}(\boldsymbol{p})$, is sampled following the practice of PIFu [31]. Different with other texture inference works [26,31] which only utilize either global or local features for texture reconstruction, we fuse *multi-scale* texture features to recover robust and detailed texture.

Geometry Reconstruction and Texture Estimation: These two modules form the core of VERTEX. They consist of three main components: Template Mapping, Template SDF Decoder and RGB Decoder. Conditioned on $\boldsymbol{z}_{shape}$, volume samples are sequentially fed to the Template Mapping and Template SDF Decoder to predict the continuous signed distance field. For texture estimation, surface points on reconstructed mesh are firstly warped to the template surface conditioned on $\boldsymbol{z}_{shape}$, and then passed through the RGB Decoder with embedding latent codes $\boldsymbol{z}_{tex}$, $\boldsymbol{z}_{loc\_tex}(\boldsymbol{p})$ and $\boldsymbol{z}_{pose}$ to predict texture.

4.2 Network Training

Based on our implicit geo-tex representation, we train the geometry and texture reconstruction network jointly. We visualize the training process in Fig. 3 and provide detailed definition of our training losses.

Data Loss: For geometry reconstruction, we mainly train by minimizing the $\ell 1$-loss between the predicted and the ground-truth point SDF values:

$$L_{geo} = \frac{1}{N_{sdf}} \sum_{i=1}^{N_{sdf}} \left\| T(W\left(\boldsymbol{p}_i, \boldsymbol{z}_{shape}\right)) - s_i \right\|_1 \tag{3}$$

where N_{sdf} represents the number of input sample points, $\boldsymbol{z}_{shape}$ is the shape latent code corresponding to the volume sample point $\boldsymbol{p}_i$, and s_i is the corresponding ground truth SDF value on the $\boldsymbol{p}_i$.

To train the texture estimation network, we minimize the $\ell 1$-loss between the regressed and the ground-truth intrinsic RGB value:

$$L_{tex} = \frac{1}{N_{sf}} \sum_{i=1}^{N_{sf}} \left\| T\left(W\left(\boldsymbol{p}_i^{(S)}, \boldsymbol{z}_{shape}\right), \boldsymbol{z}_{tex}, \boldsymbol{z}_{loc\_tex}(\boldsymbol{p}_i^{(S)})\right) - c_i \right\|_1 \tag{4}$$

where N_{sf} represents the number of input surface points, c_i is the corresponding ground truth color value on the surface point $\boldsymbol{p}_i$, and $\boldsymbol{z}_{shape}$, $\boldsymbol{z}_{tex}$ and $\boldsymbol{z}_{loc\_tex}$ are the latent codes corresponding to the $\boldsymbol{p}_i^{(S)}$.

Regularization Loss: To establish continuous mapping between the instance space and the canonical template space, we introduce an additional regularization term to constrain position offsets of points after warping:

$$L_{reg} = \frac{1}{N_{sdf}} \sum_{i=1}^{N_{sdf}} \left\| W\left(\boldsymbol{p}_i, \boldsymbol{z}_{shape}\right) - \boldsymbol{p}_i \right\|_2 \tag{5}$$

Template SDF Supervision: We supervise Template SDF Decoder directly using the sample points of the template car model. The loss is defined as:

$$L_{tp\_sdf} = \frac{1}{N_{tp\_sdf}} \sum_{i=1}^{N_{tp\_sdf}} \left\| T(\boldsymbol{p}_i^{(tp)}) - s_i^{(tp)} \right\|_1 \tag{6}$$

where N_{tp_sdf} represents the number of input sample points, $\boldsymbol{p}_i^{(tp)}$ represents the volume sample point around template model and $s_i^{(tp)}$ is the corresponding SDF value.

Overall, the total loss function is formulated as the weighted sum of above mentioned terms:

$$L = L_{tex} + w_g L_{geo} + w_{reg} L_{reg} + w_t L_{tp\_sdf} \tag{7}$$

4.3 Inference

As shown in the pipeline in Fig. 2, during inference, we first regress the signed distance field with the branch of geometry reconstruction, and then 3D points on the extracted surface are input to the branch of Texture Estimation to recover surface texture. However, because of the lack of ground truth camera intrinsic and extrinsic parameters, it is difficult for a 3D point to sample the correct local feature from feature map, which poses a significant challenge. We address the problem by setting a virtual camera and further optimizing the 6D pose under the render-and-compare optimization framework. See supplementary for details.

5 Experiments

In this section, we first introduce the new vehicle dataset in Sect. 5.1. In Sect. 5.2, we illustrate the reconstruction results under real environments and quantitative scores on our dataset compared with two state-of-art baselines. **The ablation studies and generalization to other object categories are presented in the supplementary.**

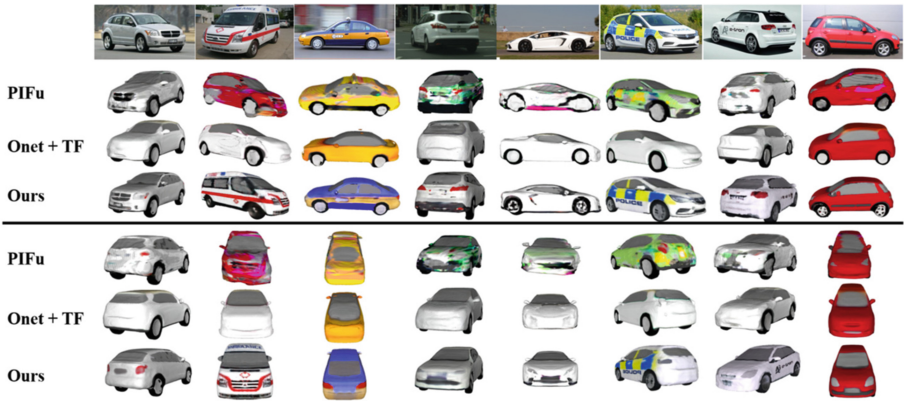

Fig. 4. Results on in-the-wild images. Monocular input images are shown in the top row. We compare 3D models reconstructed by ours and contrast works (PIFu and Onet+TF) retrained with our dataset. Two render views are provided to demonstrate reconstruction quality. Our results achieve great performance in terms of both robustness and accuracy.

5.1 Dataset

To generate synthetic dataset, we collect 83 industry-grade 3D CAD models covering common vehicle types, each of which is labeled with 23 semantic key points. We specifically select a commonly seen car as the vehicle template. To enrich the texture diversity of our dataset, we assign ten different texture for each model. We generate images with high visual fidelity using Physically Based Rendering (PBRT) [29] system and measured HDRI skymaps in the Laval HDR Sky Database [18]. Finally, we get a training set with 6300 instances and a testing set with 2000 instances in total. Please refer to supplementary for more details.

5.2 Results and Comparison

We compare our method with two state-of-the-art methods based on implicit functions. One is PIFu [31] which leverages pixel-aligned features to infer both occupied probabilities and texture distribution. The other one is Onet + Texture Field [22,26], of which Onet reconstructs shape from the monocular input image and TF infers the color for the surface points conditioned on the image and the geometry. For fair comparison, we retrain botth methods on our dataset by concatenating the RGB image and the instance mask image into a 4-channel RGB-M image as the new input. Specifically, for PIFu, instead of the stacked hourglass network [24] designed for human-related tasks, ResNet34 is set as the encoder backbone and we extract the features before every pooling layers in ResNet to obtain feature embeddings. For Onet and TF, we use the original encoder and decoder networks and adjust the dimensions of the corresponding latent codes to be equal to those in our method.

Qualitative Comparison. To prove that our method adapts to real-world images, we collect several images from Kitti [21], CityScapes [7], ApolloScape [35], CCPD[1], SCD [17] and Internet. As shown in Fig. 4, our approach generates more robust results when compared with PIFu, while recovering much more texture details than the combination of Onet and TextureField.

Table 1. Quantitative Evaluation using the FID and SSIM metrics on our dataset. For SSIM, larger is better; for FID, smaller is better. Our method achieves best in both two terms.

Method	FID ↓	SSIM ↑
PIFu*	215.8	0.6962
Onet+TF*	262.73	0.7002
Ours(w/o local feature fusion)	156.8	0.7057
Ours	**148.2**	**0.7208**
Ours(w/o joint training)	193.6	0.6902
Ours(MPV as the template)	173.2	0.6895
Ours(coupe as the template)	159.7	0.6983
Ours(sphere as the template)	187.4	0.6833

Quantitative Comparison. To quantitatively evaluate the reconstruction quality of different methods, we use two metrics: Structure similarity image metric (SSIM) [36] and Frechet inception distance (FID) [14]. These two metrics can respectively measure local and global quality of images. The SSIM is a local score that measures the distance between the rendered image and the ground truth on a per-instance basis (larger is better). FID is widely used in the GAN evaluation to evaluate perceptual distributions between a predicted image and ground truth. It is worth noting that both SSIM and FID can not evaluate the quality of generated texture of 3D objects directly. All textured 3D objects must be rendered into 2D images from the same viewpoints of ground truth. To get a more convincing result, for each generated 3D textured model, we render it from 10 different views and evaluate the scores between renderings and corresponding ground truth albedo images. As shown in Tab. 1, our method gives significantly better results in FID term and achieves state-of-the art result in SSIM term, proving that our 3D models preserve stable and fine details under multi-view observations. The quantitative results agree with the performance illustrated in qualitative comparison.

We also implement a variant of our method which does not fuse local features for the purpose of fair comparison. As shown in Tab. 1, our reconstruction conditioned on global latent codes still outperforms 'Onet+TF', demonstrating that our representation is more expressive in terms of inferring the texture on the vehicle surface.

[1] https://github.com/nicolas-gervais/predicting-car-price-from-scraped-data/tree/master/picture-scraper.

6 Conclusion

In this paper, we have introduced VERTEX, a novel method for monocular vehicle reconstruction in real-world traffic scenarios. Experiments demonstrate that our method can recover 3D vehicle models with robust and detailed texture from a monocular image. Based on the proposed implicit semantic template mapping, we have presented a new geometry-texture joint representation to constrain texture distribution on the template surface. We believe the proposed implicit geo-tex representation can further inspire 3D learning tasks on other classes of objects sharing a strong template prior.

Acknowledgements. This paper is supported by the National Key Research and Development Program of China [2018YFB2100500].

References

1. Beker, D., et al.: Monocular differentiable rendering for self-supervised 3D object detection (2020)
2. Carr, J.C., Beatson, R.K., Cherrie, J.B., Mitchell, T.J., Evans, T.R.: Reconstruction and representation of 3D objects with radial basis functions. In: Computer Graphics (2001)
3. Chan, E.R., Monteiro, M., Kellnhofer, P., Wu, J., Wetzstein, G.: pi-GAN: periodic implicit generative adversarial networks for 3D-aware image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5799–5809 (2021)
4. Chang, A.X., et al.: An information-rich 3D model repository. Comput. Sci. (2015)
5. Chen, Z., Zhang, H.: Learning implicit fields for generative shape modeling. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
6. Choy, C.B., Xu, D., Gwak, J.Y., Chen, K., Savarese, S.: 3D-R2N2: A Unified Approach for Single and Multi-view 3D object reconstruction. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 628–644. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_38
7. Cordts, M., et al.: The cityscapes dataset for semantic urban scene understanding. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3213–3223 (2016)
8. Deng, Y., Yang, J., Tong, X.: Deformed implicit field: Modeling 3D shapes with learned dense correspondence. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10286–10296 (2021)
9. Deng, Y., Yang, J., Xiang, J., Tong, X.: Gram: generative radiance manifolds for 3D-aware image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10673–10683 (2022)
10. Goel, S., Kanazawa, A., Malik, J.: Shape and viewpoint without keypoints. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12360, pp. 88–104. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58555-6_6

11. Gropp, A., Yariv, L., Haim, N., Atzmon, M., Lipman, Y.: Implicit geometric regularization for learning shapes. arXiv:2002.10099 (2020)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
13. Henderson, P., Tsiminaki, V., Lampert, C.: Leveraging 2D data to learn textured 3D mesh generation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
14. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: GANs trained by a two time-scale update rule converge to a local NASH equilibrium. In: Advances in Neural Information Processing Systems, pp. 6626–6637 (2017)
15. Kaiming, H., Georgia, G., Piotr, D., Ross, G.: Mask R-CNN. IEEE Trans. Pattern Anal. Mach. Intell, pp. 1–1 (2017)
16. Kanazawa, A., Tulsiani, S., Efros, A.A., Malik, J.: Learning category-specific mesh reconstruction from image collections. In: Proceedings of the European Conference on Computer Vision (ECCV) (2018)
17. Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3D object representations for fine-grained categorization. In: 4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13). Sydney, Australia (2013)
18. Lalonde, J.F,et al.: The Laval HDR sky database. http://sky.hdrdb.com (2016)
19. Li, W., et al.: AADS: Augmented autonomous driving simulation using data-driven algorithms. Science Robotics 4 (2019)
20. Meng, D., et al.: Parsing-based view-aware embedding network for vehicle re-identification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) June 2020
21. Menze, M., Heipke, C., Geiger, A.: Object scene flow. ISPRS J. Photogrammetry Remote Sens.(JPRS) (2018)
22. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: learning 3D reconstruction in function space. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
23. Miao, H., Lu, F., Liu, Z., Zhang, L., Manocha, D., Zhou, B.: Robust 2D/3D vehicle parsing in CVIS (2021)
24. Newell, A., Yang, K., Deng, J.: Stacked Hourglass Networks for Human Pose Estimation. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 483–499. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_29
25. Niemeyer, M., Geiger, A.: Giraffe: representing scenes as compositional generative neural feature fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11453–11464 (2021)
26. Oechsle, M., Mescheder, L., Niemeyer, M., Strauss, T., Geiger, A.: Texture fields: learning texture representations in function space. In: Proceedings IEEE International Conf. on Computer Vision (ICCV) (2019)
27. Park, E., Yang, J., Yumer, E., Ceylan, D., Berg, A.C.: Transformation-grounded image generation network for novel 3D view synthesis. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
28. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: DeepSDF: Learning continuous signed distance functions for shape representation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) June 2019
29. Pharr, M., Jakob, W., Humphreys, G.: Physically based rendering: from theory to implementation. Morgan Kaufmann (2016)

30. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
31. Saito, S., Huang, Z., Natsume, R., Morishima, S., Li, H., Kanazawa, A.: PIFU: pixel-aligned implicit function for high-resolution clothed human digitization. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV) (2019)
32. Shen, C., O"Brien, J.F., Shewchuk, J.R.: Interpolating and approximating implicit surfaces from polygon soup. ACM Trans. Graph. **23**(3), pp. 896–904 (2004)https://doi.org/10.1145/1186562.1015816
33. Sun, Y., Liu, Z., Wang, Y., Sarma, S.E.: Im2avatar: colorful 3D reconstruction from a single image (2018)
34. Turk, G., O'Brien, J.F.: Modelling with implicit surfaces that interpolate. ACM Trans. Graph. **21**(4), 855–873 (2002)
35. Wang, P., Huang, X., Cheng, X., Zhou, D., Geng, Q., Yang, R.: The apolloscape open dataset for autonomous driving and its application. IEEE Trans. pattern. Anal. Mach. Intell (2019)
36. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)
37. Xu, Q., Wang, W., Ceylan, D., Mech, R., Neumann, U.: DISN: Deep implicit surface network for high-quality single-view 3D reconstruction. In: Advances in Neural Information Processing Systems 32 (2019)
38. Xu, Y., Peng, S., Yang, C., Shen, Y., Zhou, B.: 3D-aware image synthesis via learning structural and textural representations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 18430–18439 (2022)
39. Zheng, Z., Yu, T., Dai, Q., Liu, Y.: Deep implicit templates for 3D shape representation (2020)
40. Zhu, J.Y., et al.: Visual object networks: Image generation with disentangled 3D representations. In: Advances in Neural Information Processing Systems 31 (2018)