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

Xiaochen Zhao¹, Zerong Zheng¹, Chaonan Ji¹, Zhenyi Liu², Siyou Lin¹, Tao Yu¹, Jinli Suo¹, Yebin Liu¹ ¹Tsinghua University, Beijing, China ²Jilin University, Jilin, 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 semanticpreserving 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.

1. Introduction

Monocular visual scene understanding is a fundamental technology for many automatic applications, especially in the field of autonomous driving. Using only a singleview driving image, available vehicle parsing studies have covered popular topics starting from 2D vehicle detection [2, 37, 35, 14, 33], then 6D vehicle pose recovery [62, 38, 28, 56, 12, 13, 3, 36], and finally vehicle shape reconstruction [30, 54, 22, 27, 15, 39, 64]. 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 [34], vehicle tracking [40], vehicle parsing [43] and so on.



Figure 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 [48, 42]. Texture fields (TF) [46] and PIFu [52] 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 [65, 11] in a semantic-preserving manner. In our geotex 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. [65, 11] 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 [52, 46, 22]. 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 geotex 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, partlevel understanding and vehicle editing;

- a joint training strategy leveraging the consistency between RGB color and part semantics for semanticspreserving 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

In the field of autonomous driving, works for shape recovery and pose estimation [31, 54, 39, 64, 30, 17, 5, 44] can be naively extended to texture reconstruction with projective texturing. However, direct unprojection can only obtain the texture of visible parts and is incapable of recovering consistent 3D texture.

Recently, many works [1, 22, 27, 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 [7], many works [46, 55, 9] train deep neural networks to perform vehicle reconstruction from images. Based on volumentrically representation like 3D voxel [55] and implicit functions [46], 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 [67, 50, 66, 47] 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 [4, 57, 53]. In recent years, implicit functions have been implemented with neural networks [48, 42, 8, 18, 63, 52, 25, 6, 20] and have shown promising results. For example, DeepSDF [48] 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 [42, 8, 63, 52].

As for texture inference, both TF [46] and PIFu [52] define texture implicitly as a function of 3D positions. The



Figure 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.

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 [52, 46] 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.

2.3. Warping based Geometry Template Learning

As one of the most recent research hotpots, warping based geometry can be divided into mesh-based methods [58, 61, 19, 24] and implicit methods [24, 19], according to different representations. [58, 61] generate various objects by deforming Ellipsoid meshes using graph CNN, while [24, 19] use neural ODEs for learned shape deformations instead. Recent works of [65, 11] argued for implicitly learning templates for a collection of shapes and establishing dense correspondences. However, without taking texture information into account, the learned geometry templates and the mapping ignore semantic alignment as shown in Fig. 8. In contrast, our representation presents the semantic template and firstly exploits implicit template prior for monocular 3D reconstruction, achieving good performance.

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

State-of-the-art deep implicit representations for 3D objects, such as PIFu and TF, represent texture and geometry using *separate* implicit fields. However, geometry and texture are never fully disentangled in either PIFu or TF, as they rely on the location of surface to diffuse the color into

3D space. It leads to the fact that each texture field can only corresponds to one specific surface. This could be problematic when surface geometry is not available and has to be inferred. If the inferred geometry is slightly different from the ground-truth, the surface texture extracted from the texture field could be erroneous.

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 [65] 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 S_T as the level set of a signed distance function $F : \mathbb{R}^3 \mapsto \mathbb{R}$, i.e. F(q) = 0, where $q \in \mathbb{R}^3$ denotes a 3D point. Then our representation can be formulated as:

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

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

3.2. Formulation for Image-based Reconstruction

For a specific instance, the shape information is defined by z_{shape} , while the texture information is encoded as 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 $z_{loc.tex}(p)$ 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:

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

where $T : S_T \times \mathcal{X}_{tex} \times \mathcal{X}_{loc_tex} \mapsto \mathbb{R}^3$ is conditioned on the latent codes $z_{tex} and z_{loc_tex}$.

Compared with the previous works, the main advantage of our joint representation is that it explicitly constrains the texture distribution on the 2D surface of the template model, which effectively reduces the complexity of regressing texture. Besides, with the template being an intermediary, shape latent codes and texture latent codes are well decoupled. As a result, it is easy to combine different pairs of latent codes to transfer texture across shapes, as demonstrated in Fig. 6. Moreover, template can be custom-designed to assign extra semantics, such as material information. Observing that vehicles always share similar material in corresponding parts (e.g. glass in car window, metal in car body), our representation can become a promising solution to monocular vehicle material recovery.

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 Section 4.

4. Joint Geo-tex Training Method

In this section, we first describe our network architecture in Sec. 4.1. In Sec. 4.2, we present how we train our geometry reconstruction network and texture estimation network jointly. The inference scheme is presented in Sec. 4.3.

4.1. Network Architecture

Fig. 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 offthe-shelf 2D detectors [26], 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 [51], 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, $z_{shape} \& z_{tex}$, are extracted from the input image and recovered albedo map using two separate ResNetbased [21] encoders respectively. The local texture feature, $z_{loc.tex}(p)$, is sampled following the practice of PIFu [52]. Different with other texture inference works [52, 46] 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 & 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 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 z_{shape} , and then passed through the RGB Decoder with embedding latent codes z_{tex} , $z_{loc.tex}(p)$ and 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. In this way, we are able to leverage the consistency between RGB color and semantic part segmentation to force the template mapping to be semantic-preserving. 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 ℓ 1-loss between the predicted and the ground-truth point SDF values:

$$L_{geo} = \frac{1}{N_{sdf}} \sum_{i=1}^{N_{sdf}} \|T(W(\boldsymbol{p}_{i}, \boldsymbol{z}_{shape})) - s_{i}\|_{1} \quad (3)$$

where N_{sdf} represents the number of input sample points, z_{shape} is the shape latent code corresponding to the volume sample point p_i , and s_i is the corresponding ground truth SDF value on the p_i .

To train the texture estimation network, we minimize the ℓ 1-loss between the regressed and the ground-truth intrinsic



Figure 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. Each neural network module plays the role of domain mapping conditioned on corresponding latent codes as mentioned in overview. Note that Z in RGB Decoder is the concatenation of the global and local texture latent codes.

RGB value:

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

$$(4)$$

where N_{sf} represents the number of input surface points, c_i is the corresponding ground truth color value on the surface point p_i , and z_{shape} , z_{tex} and z_{loc_tex} are the latent codes corresponding to the $p_i^{(S)}$.

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:

$$L_{reg} = \frac{1}{N_{sdf}} \sum_{i=1}^{N_{sdf}} ||W(\boldsymbol{p}_{i}, \boldsymbol{z}_{shape}) - \boldsymbol{p}_{i}||_{2}$$
(5)

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

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

where N_{tp_sdf} represents the number of input sample points, $p_i^{(tp)}$ 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:

$$L = L_{tex} + w_g L_{geo} + w_{reg} L_{reg} + w_t L_{tp_sdf}$$
(7)

With embedding latent codes implicitly depending on the parameters of encoders, the whole network is trained end-to-end by minimizing Eq. 7. See supplementary for implementation details.

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 Sec. 5.1. In Sec. 5.2, we illustrate the reconstruction results under real environments and quantitative scores on our dataset compared with two state-of-art baselines. For evaluation in Sec. 5.3, we conducted ablation studies. Finally, we show results on other object categories to prove representation generalization in Sec. 5.4. More experimental details are presented in the supplementary.

5.1. Dataset

To generate synthetic dataset, we collect 83 industrygrade 3D CAD models covering common vehicle types, each of which is labeled with 23 semantic key points. These key points pairs contain semantic correspondences and are served to evaluate the accuracy of semantic correspondence in our experiments. 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. To simulate the driving view in real street environment, car models are randomly rotated and placed in different 3D locations, and then rendered in high-resolution (2048×1024) and wide-angle ($fov = 50^{\circ}$) image. We generate images with high visual fidelity using Physically Based Rendering (PBRT) [49] system and measured HDRI skymaps in the



Figure 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.

Laval HDR Sky Database [32]. 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.

As for the supervision for geometry reconstruction, we use the same data preparation method as Onet [42] to generate watertight meshes and follow the sampling strategy in DeepSDF [48] to obtain spatial points with their calculated signed distance value.

5.2. Results and Comparison

We compare our method with two state-of-the-art methods based on implicit functions. One is PIFu [52] which leverages pixel-aligned features to infer both occupied probabilities and texture distribution. The other one is Onet + Texture Field [42, 46], 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 [45] 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

Method	$FID\downarrow$	SSIM \uparrow
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

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.

Kitti [41], CityScapes [10], ApolloScape [59], CCPD¹, SCD [29] 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.

Quantitative Comparison. To quantitatively evaluate the reconstruction quality of different methods, we use two metrics: Structure similarity image metric (SSIM) [60] and Frechet inception distance (FID) [23]. 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

¹https://github.com/nicolas-gervais/predicting-car-price-from-scrapeddata/tree/master/picture-scraper



Figure 5: Illumination Removal and Material Analysis. Conditioning texture inference with the predicted albedo maps improves the reconstruction robustness. Then, benefiting from our implicit semantic template mapping, we can assign material information from the pre-designed template for the reconstructed model and then render realistic images. Note that different specular reflections shown in right cols are caused by material differences (e.g. metal material body and glass material window), proving our diverse material identification.



Figure 6: Texture Transfer. By extracting shape latent codes from top row and texture latent codes from left col, our representation can freely couple shape and texture latent codes to generate new vehicle instances.

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.

5.3. Evaluation

Evaluation on the disentanglement of representation. With the incorporation of the implicit semantic template mapping, our model avoid entangled representations and recover the surface texture in the 2-manifold of surfaces defined in the canonical template filed, thus supporting many downstream tasks such as texture transfer and editing. As shown in Fig. 6, we extract shape latent codes from top row to reconstruct the geometry and extract texture latent codes from the left col to recover the surface texture, resulting in plausible texture transfer, which proves the practicality of our disentanglement of geometry and texture latent spaces.

Evaluation on the joint training strategy. Our method is able to establish semantic correspondences between different vehicle instances (right part of Fig. 3), attributed in our joint training strategy. We retrain a baseline network by firstly train the geometry branch and then train the texture branch conditioned on a fixed template mapping. As shown in Fig. 8, without leveraging texture information as the guidance, the mapping process follows the principle of shortest distance to establish correspondences and ignores semantic information. To evaluate the accuracy of semantic mapping, we utilize the key points annotation in our dataset and calculate the distance errors between the mapped key points and the target ones. RMSE score for comparison network is 5.259 while ours is 0.7594, which demonstrates that the joint training strategy helps establish meaningful semantic dense correspondences between various instances. Moreover, the decrease of numerical result in Tab. 1 indicates that texture reconstruction quality does benefit from the semantic template mapping, implemented by geo-tex joint training strategy.

Evaluation on illumination removal and material identification. To alleviate the lighting effects of image appearance, we add an image-translation network to convert the input color images to albedo maps. The module effectively helps our network remove illumination and shading effects in 2D image domain and contributes to robust texture results. We retrain a comparison network by directly feeding original color images into texture encoders. As shown



Figure 7: Representation Generalization. Our model is extended to other object categories, with templates obtained from DIT.



Figure 8: Joint training strategy contributes to semantic template mapping. Note the semantic misalignment of front/back windows in the right column.

in Fig. 5, the network without the module tends to generate noisy results. Furthermore, though material identification based on monocular image is an ill-posed problem, with our implicit semantic template, the reconstructed intrinsic textured models obtain material parameters from the predesigned template model and are able to generate realistic renderings through model relighting, as shown in the right part of the Fig. 5.

Evaluation on the choice of the template. While our method select a sedan serving as the template car, to explore the sensitivity of our method to the choice of the template, we conduct three comparison experiments choosing an MPV, coupe and unit sphere as the template separately. For ease of comparison, we do not fuse local information for these experiments. Quantitative results are presented in Tab. 1. In general, our method is relatively insensitive to the template model and able to generate meaningful reconstruction results with different types of templates. We analyze that this arises from the fact that car shapes are almost homomorphic to the sphere, hence dense correspondences can be established for these template surfaces. Specifically, choosing the model close to the mean mesh within the category as the template will cause better performance.

Evaluation on topology-varying reconstruction. As the template mapping operator is defined in an implicit manner, our method preserves the advantage of implicit functions to represent topology-varying objects. Fig. 9 presents an example of the reconstruction of a car with a separate rear spoiler. Results on other objects categories (Fig. 7) illustrates reconstruction cases with genus, the topology of which differs from the template meshes.



Figure 9: Topology-varying vehicle reconstruction. Our method is able to represent vehicles with various typologies; see the rear spoiler of the reconstructed model.

5.4. Representation Generalization

In this section, we extend our representation to other object categories. We separately train our model on "sofa" and "airplane" category from ShapeNet [7]. To further prove our representation power, we also experiment on a collected bicycle dataset containing about 200 shapes. We use the template models learned by DIT [65] and conduct single-image 3D reconstruction experiments on these synthetic datasets. As shown in Fig. 7, our representation is qualified for generalization to other shape categories.

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, and have shown how to implement it with joint training strategy and a novel dataset. Moreover, we have demonstrated the advantages brought by the implicit semantic template to latent space disentanglement and material identification. We believe the proposed implicit geo-tex representation can further inspire 3D learning tasks on other classes of objects sharing a strong template prior. In future, we plan to extend our framework to handle the task of monocular video based vehicle reconstruction and leverage temporal information to improve the accuracy of texture estimation.

References

- Deniz Beker, Hiroharu Kato, Mihai Adrian Morariu, Takahiro Ando, Toru Matsuoka, Wadim Kehl, and Adrien Gaidon. Monocular differentiable rendering for selfsupervised 3d object detection, 2020. 2
- [2] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection, 2020. 1
- [3] Ming Cai and Ian Reid. Reconstruct locally, localize globally: A model free method for object pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2020. 1
- [4] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, and T. R. Evans. Reconstruction and representation of 3d objects with radial basis functions. In *Computer Graphics*, 2001. 2
- [5] Florian Chabot, Mohamed Chaouch, Jaonary Rabarisoa, Celine Teuliere, and Thierry Chateau. Deep manta: A coarseto-fine many-task network for joint 2d and 3d vehicle analysis from monocular image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), July 2017. 2
- [6] Rohan Chabra, J. E. Lenssen, Eddy Ilg, Tanner Schmidt, J. Straub, S. Lovegrove, and Richard A. Newcombe. Deep local shapes: Learning local sdf priors for detailed 3d reconstruction. In *ECCV*, 2020. 2
- [7] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, and Hao Su. Shapenet: An informationrich 3d model repository. *Computer Science*, 2015. 2, 8
- [8] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
 2
- [9] Christopher B. Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 628–644, Cham, 2016. Springer International Publishing. 2
- [10] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, M. Enzweiler, Rodrigo Benenson, Uwe Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 3213– 3223, 2016. 6
- [11] Yu Deng, Jiaolong Yang, and Xin Tong. Deformed implicit field: Modeling 3d shapes with learned dense correspondence. *CoRR*, abs/2011.13650, 2020. 2, 3
- [12] T Do, Trung Pham, Ming Cai, and Ian Reid. Real-time monocular object instance 6d pose estimation. 2019. 1
- [13] Thanh-Toan Do, Ming Cai, Trung Pham, and Ian Reid. Deep-6dpose: Recovering 6d object pose from a single rgb image, 2018. 1
- [14] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 1

- [15] F. Engelmann, J. Stückler, and B. Leibe. Samp: Shape and motion priors for 4d vehicle reconstruction. In 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), pages 400–408, 2017. 1
- [16] Shubham Goel, Angjoo Kanazawa, , and Jitendra Malik. Shape and viewpoints without keypoints. In ECCV, 2020.
 2
- [17] Alexander Grabner, Peter M. Roth, and Vincent Lepetit. 3d pose estimation and 3d model retrieval for objects in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2
- [18] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Y. Lipman. Implicit geometric regularization for learning shapes. *ArXiv*, abs/2002.10099, 2020. 2
- [19] Kunal Gupta and Manmohan Chandraker. Neural mesh flow: 3d manifold mesh generationvia diffeomorphic flows. arXiv preprint arXiv:2007.10973, 2020. 3
- [20] Zekun Hao, Hadar Averbuch-Elor, Noah Snavely, and Serge Belongie. Dualsdf: Semantic shape manipulation using a two-level representation. Arxiv, 2020. 2
- [21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016. 4
- [22] Paul Henderson, Vagia Tsiminaki, and Christoph Lampert. Leveraging 2D data to learn textured 3D mesh generation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 2
- [23] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In Advances in neural information processing systems, pages 6626–6637, 2017. 6
- [24] Chiyu Jiang, Jingwei Huang, Andrea Tagliasacchi, Leonidas Guibas, et al. Shapeflow: Learnable deformations among 3d shapes. arXiv preprint arXiv:2006.07982, 2020. 3
- [25] Chiyu Jiang, Avneesh Sud, Ameesh Makadia, Jingwei Huang, and Thomas Funkhouser. Local implicit grid representations for 3d scenes. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020. 2
- [26] He Kaiming, Gkioxari Georgia, Dollar Piotr, and Girshick Ross. Mask r-cnn. *IEEE Transactions on Pattern Analysis* and Machine Intelligence, PP:1–1, 2017. 4
- [27] Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 1, 2
- [28] Wadim Kehl, Fabian Manhardt, Federico Tombari, Slobodan Ilic, and Nassir Navab. Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In *Proceedings* of the IEEE International Conference on Computer Vision (ICCV), Oct 2017. 1
- [29] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In 4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13), Sydney, Australia, 2013. 6

- [30] Jason Ku, Alex D. Pon, and Steven L. Waslander. Monocular 3d object detection leveraging accurate proposals and shape reconstruction. In *Proceedings of the IEEE/CVF Conference* on Computer Vision and Pattern Recognition (CVPR), June 2019. 1, 2
- [31] Abhijit Kundu, Yin Li, and James M. Rehg. 3d-rcnn: Instance-level 3d object reconstruction via render-andcompare. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018. 2
- [32] J.-F. Lalonde, L.-P. Asselin, J. Becirovski, Y. Hold-Geoffroy, M. Garon, M.-A. Gardner, and J. Zhang. The laval hdr sky database. http://www.tp-ontrol.hu/index. php/TP_Toolbox, 2016. 6
- [33] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *Proceedings of the European Confer*ence on Computer Vision (ECCV), September 2018. 1
- [34] Wei Li, Chengwei Pan, Rong Zhang, Jiaping Ren, Yuexin Ma, Jin Fang, Feilong Yan, Qichuan Geng, Xinyu Huang, Huajun Gong, Weiwei Xu, Guoping Wang, Dinesh Manocha, and Ruigang Yang. Aads: Augmented autonomous driving simulation using data-driven algorithms. *Science Robotics*, 4, 2019. 1
- [35] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 1
- [36] Jin Liu and Sheng He. 6d object pose estimation based on 2d bounding box, 2019. 1
- [37] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector. *Lecture Notes* in Computer Science, page 21–37, 2016. 1
- [38] Y. Lu, S. Kourian, C. Salvaggio, C. Xu, and G. Lu. Single image 3d vehicle pose estimation for augmented reality. In 2019 IEEE Global Conference on Signal and Information Processing (GlobalSIP), pages 1–5, 2019. 1
- [39] Fabian Manhardt, Wadim Kehl, and Adrien Gaidon. Roi-10d: Monocular lifting of 2d detection to 6d pose and metric shape. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2069–2078, 2019. 1, 2
- [40] Dechao Meng, Liang Li, Xuejing Liu, Yadong Li, Shijie Yang, Zheng-Jun Zha, Xingyu Gao, Shuhui Wang, and Qingming Huang. 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. 1
- [41] Moritz Menze, Christian Heipke, and Andreas Geiger. Object scene flow. *ISPRS Journal of Photogrammetry and Remote Sensing (JPRS)*, 2018. 6
- [42] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019. 1, 2, 6
- [43] Hui Miao, Feixiang Lu, Zongdai Liu, Liangjun Zhang, Dinesh Manocha, and Bin Zhou. Robust 2d/3d vehicle parsing in cvis, 2021. 1

- [44] Roozbeh Mottaghi, Yu Xiang, and Silvio Savarese. A coarseto-fine model for 3d pose estimation and sub-category recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 2
- [45] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. arXiv e-prints, 2016. 6
- [46] Michael Oechsle, Lars Mescheder, Michael Niemeyer, Thilo Strauss, and Andreas Geiger. Texture fields: Learning texture representations in function space. In *Proceedings IEEE International Conf. on Computer Vision (ICCV)*, 2019. 1, 2, 3, 4, 6
- [47] Eunbyung Park, Jimei Yang, Ersin Yumer, Duygu Ceylan, and Alexander C. Berg. Transformation-grounded image generation network for novel 3d view synthesis. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017. 2
- [48] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 2, 6
- [49] Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically based rendering: From theory to implementation*. Morgan Kaufmann, 2016. 5
- [50] K. Rematas, C. H. Nguyen, T. Ritschel, M. Fritz, and T. Tuytelaars. Novel views of objects from a single image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(8):1576–1590, 2017. 2
- [51] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. Unet: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 4
- [52] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Hao Li, and Angjoo Kanazawa. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019. 1, 2, 3, 4, 6
- [53] Chen Shen, James F. O"Brien, and Jonathan R. Shewchuk. Interpolating and approximating implicit surfaces from polygon soup. *Acm Transactions on Graphics*, 23(3):896–904, 2004. 2
- [54] Xibin Song, Peng Wang, Dingfu Zhou, Rui Zhu, Chenye Guan, Yuchao Dai, Hao Su, Hongdong Li, and Ruigang Yang. Apollocar3d: A large 3d car instance understanding benchmark for autonomous driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2019. 1, 2
- [55] Yongbin Sun, Ziwei Liu, Yue Wang, and Sanjay E. Sarma. Im2avatar: Colorful 3d reconstruction from a single image, 2018. 2
- [56] Martin Sundermeyer, Zoltan-Csaba Marton, Maximilian Durner, Manuel Brucker, and Rudolph Triebel. Implicit 3d orientation learning for 6d object detection from rgb images. In Proceedings of the European Conference on Computer Vision (ECCV), September 2018. 1

- [57] GREG TURK and JAMES F. O'BRIEN. Modelling with implicit surfaces that interpolate. Acm Transactions on Graphics, 21(4):p.855–873, 2002. 2
- [58] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. Pixel2mesh: Generating 3d mesh models from single rgb images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 52–67, 2018. 3
- [59] Peng Wang, Xinyu Huang, Xinjing Cheng, Dingfu Zhou, Qichuan Geng, and Ruigang Yang. The apolloscape open dataset for autonomous driving and its application. *IEEE* transactions on pattern analysis and machine intelligence, 2019. 6
- [60] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 6
- [61] Chao Wen, Yinda Zhang, Zhuwen Li, and Yanwei Fu. Pixel2mesh++: Multi-view 3d mesh generation via deformation. In *ICCV*, 2019. 3
- [62] Di Wu, Zhaoyong Zhuang, Canqun Xiang, Wenbin Zou, and Xia Li. 6d-vnet: End-to-end 6-dof vehicle pose estimation from monocular rgb images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2019. 1
- [63] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 492–502. Curran Associates, Inc., 2019. 2
- [64] Sergey Zakharov, Wadim Kehl, Arjun Bhargava, and Adrien Gaidon. Autolabeling 3d objects with differentiable rendering of sdf shape priors. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020. 1, 2
- [65] Zerong Zheng, Tao Yu, Qionghai Dai, and Yebin Liu. Deep implicit templates for 3d shape representation, 2020. 2, 3, 8
- [66] Tinghui Zhou, Shubham Tulsiani, Weilun Sun, Jitendra Malik, and Alexei A. Efros. View synthesis by appearance flow. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 286–301, Cham, 2016. Springer International Publishing. 2
- [67] Jun-Yan Zhu, Zhoutong Zhang, Chengkai Zhang, Jiajun Wu, Antonio Torralba, Josh Tenenbaum, and Bill Freeman. Visual object networks: Image generation with disentangled 3d representations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, Advances in Neural Information Processing Systems, volume 31, pages 118–129. Curran Associates, Inc., 2018. 2