GSNet: Joint Vehicle Pose and Shape Reconstruction with Geometrical and Scene-aware Supervision

Lei Ke[1], Shichao Li[1], Yanan Sun[1], Yu-Wing Tai[1,2], and Chi-Keung Tang[1]

[1] The Hong Kong University of Science and Technology
[2] Kwai Inc.
{lkeab,slicd,ysuncd,yuwing,cktang}@cse.ust.hk

Abstract. We present a novel end-to-end framework named as GSNet (**G**eometric and **S**cene-aware **Net**work), which jointly estimates 6DoF poses and reconstructs detailed 3D car shapes from single urban street view. GSNet utilizes a unique four-way feature extraction and fusion scheme and directly regresses 6DoF poses and shapes in a single forward pass. Extensive experiments show that our diverse feature extraction and fusion scheme can greatly improve model performance. Based on a divide-and-conquer 3D shape representation strategy, GSNet reconstructs 3D vehicle shape with great detail (1352 vertices and 2700 faces). This dense mesh representation further leads us to consider geometrical consistency and scene context, and inspires a new multi-objective loss function to regularize network training, which in turn improves the accuracy of 6D pose estimation and validates the merit of jointly performing both tasks. We evaluate GSNet on the largest multi-task ApolloCar3D benchmark and achieve state-of-the-art performance both quantitatively and qualitatively. Project page is available at https://lkeab.github.io/gsnet/.

Keywords: Vehicle Pose and Shape Reconstruction; 3D Traffic Scene Understanding

1 Introduction

Traffic scene understanding is an active area in autonomous driving, where one emerging and challenging task is to perceive 3D attributes (including translation, rotation and shape) of vehicle instances in a dynamic environment as Figure 1 shows. Compared to other scene representations such as 2D/3D bounding boxes [5,27,37], semantic masks [7,40] and depth maps [60], representing traffic scene with 6D object pose and detailed 3D shape is more informative for spatial reasoning and motion planning of self-driving cars.

Due to the lack of depth information in monocular RGB images, many existing works resort to stereo camera rigs [27,28] or expensive LiDAR [62,63,21]. However, they are limited by constrained perception range [27] or sparse 3D points for distant regions in the front view [48]. When using only a single RGB image, works that jointly reconstruct vehicle pose and shape can be classified

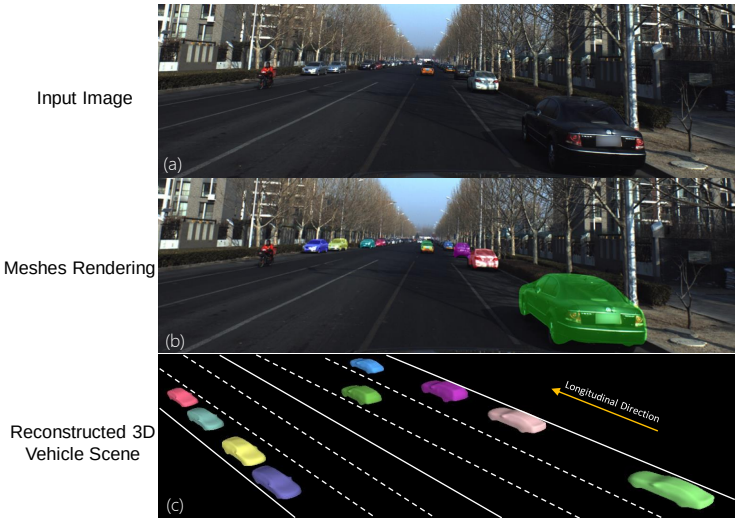

Fig. 1. Joint vehicle pose and shape reconstruction results of our GSNet, where (a) is the input RGB image, (b) shows the reconstructed 3D car meshes projected onto the original image, (c) is a novel aerial view of the reconstructed 3D traffic scene. Corresponding car instances in (b) and (c) are depicted in the same color.

into two categories: *fitting-based* and direct *regression-based*. *Fitting-based* methods [3,49,48] use a two-stage strategy where they first extract 2D image cues such as bounding boxes and keypoints and then fit a 3D template vehicle to best match its 2D image observations. The second stage is a post-processing step that is usually time-consuming due to iterative non-linear optimization, making it less applicable for real-time autonomous driving. On the contrary, *regression-based* methods [22,48] directly predict 3D pose/shape parameters with a single efficient forward pass of a deep network and is gaining increasing popularity with the growing scale of autonomous driving datasets.

Despite the recent regression-based methods having achieved remarkable performance for joint vehicle pose estimation and 3D shape reconstruction, we point out some unexplored yet valuable research questions: (1) Most regression-based networks [22,48,26] inherit classical 2D object detection architectures that solely use region of interest (ROI) features to regress 3D parameters. *How other potential feature representation can improve network performance* is less studied. (2) Deep networks require huge amounts of supervision [18], where useful supervisory signals other than manually annotated input-target pairs are favorable. Consistency brought by projective geometry is one possibility, yet the optimal design is still under-explored. Render-and-compare loss was used in [22] but it suffers from ambiguities where similar 2D projected masks can correspond to different 3D unknown parameters. For example, a mask similar to the ground truth mask is produced after changing the ground truth 3D pose by 180 degrees around the symmetry axis, i.e., the prediction is not penalized enough despite

being incorrect. (3) Previous regression-based works only penalize prediction error for single car instance and separate it from its environmental context, but a traffic scene includes the interaction between multiple instances and the relationship between instances with the physical world. We argue that considering these extra information can improve the training of a deep network.

We investigate these above questions and propose GSNet (**G**eometric and **S**cene-aware **Net**work), an end-to-end multi-task network that can estimate 6DoF car pose and reconstruct dense 3D shape simultaneously. We go beyond the ROI features and systematically study how other visual features that encode geometrical and visibility information can improve the network performance, where a simple yet effective four-way feature fusion scheme is adopted. Equipped with a dense 3D shape representation achieved by a *divide-and-conquer* strategy, we further design a multi-objective loss function to effectively improve network learning as validated by extensive experiments. This loss function considers geometric consistency using the projection of 66 semantic keypoints instead of masks which effectively reduces the ambiguity issue. It also incorporates a scene-aware term considering both inter-instance and instance-environment constraints.

In summary, our contributions are: (1) A novel end-to-end network that can jointly reconstruct 3D pose and dense shape of vehicles, achieving state-of-the-art performance on the largest multi-task ApolloCar3D benchmark [48]. (2) We propose an effective approach to extract and fuse diverse visual features, where systematic ablation study is shown to validate its effectiveness. (3) GSNet reconstructs fine-grained 3D meshes (1352 vertices) by our *divide-and-conquer* shape representation for vehicle instances rather than just 3D bounding boxes, wireframes [67] or retrieval [3,48]. (4) We design a new hybrid loss function to promote network performance, which considers both geometric consistency and scene constraints. This loss is made possible by the dense shape reconstruction, which in turn promotes the 6D pose estimation precision and sheds light on the benefit of jointly performing both tasks.

2 Related Work

Monocular 6DoF pose estimation. Traditionally, 6D object pose estimation is handled by creating correspondences between the objects known 3D model and 2D pixel locations, followed by Perspective-n-Point (PnP) algorithm [45,54,39]. For recent works, [2,13] construct templates and calculate the similarity score to obtain the best matching position on the image. In [38,45,59], 2D regional image features are extracted and matched with the features on 3D model to establish 2D-3D relation which thus require sufficient textures for matching. A single-shot deep CNN is proposed in [51] which regresses 6D object pose in one stage while in [50] a two-stage method is used: 1) SSD [35] for detecting bounding boxes and identities; 2) augmented autoencoder predicts object rotation using domain randomization [52]. Most recently, Hu et al. [14] introduces a segmentation-based method by combining local pose prediction from each visible part of the objects. Comparing to the cases in self-driving scenarios, these methods [16,59,42] are

applied to indoor scenes with a small variance in translation especially along the longitudinal axis. Although using keypoints information, our model does *not* treat pose estimation as a PnP problem and is trained end-to-end.

Monocular 3D shape reconstruction. With the advent of large-scale shape datasets [4] and the progress of data-driven approaches, 3D shape reconstruction from a single image based on convolutional neural networks is drawing increasing interests. Most of these approaches [43,56,66,61,47,44,19,30] focus on general objects in the indoor scene or in the wild [15], where single object is shot in a close distance and occupies the majority of image area. Different from them, GSNet is targeted for more complicated traffic environment with far more vehicle instances to reconstruct per image, where some of them are even under occlusion at a long distance (over 50 meters away).

Joint vehicle pose and shape reconstruction. 3D traffic scene understanding from a single RGB image is drawing increasing interests in recent years. However, many of these approaches only predict object orientation with 3D bounding boxes [6,29,62,33,1,46,60]. When it comes to 3D vehicle shape reconstruction, since the KITTI dataset [11] labels cars using 3D bounding boxes with no detailed 3D shape annotation, existing works mainly use wireframes [67,26,55,20] or retrieve from CAD objects [3,48,36,57]. In [64], the authors utilize 3D wireframe vehicle models to jointly estimate multiple objects in a scene and find that more detailed representations of object shape are highly beneficial to 3D scene understanding. DeepMANTA [3] adopts a coarse-to-fine refinement strategy to first regress 2D bounding box positions and generate 3D bounding boxes and finally obtain pose estimation results via 3D template fitting [25] by using the matched skeleton template to best fit its 2D image observations, which requires no image with 3D ground truth. *Most related to ours*, 3D-RCNN [22] regresses 3D poses and deformable shape parameters in a single forward pass, but it uses coarse voxel shape representation and the proposed render-and-compare loss causes ambiguity during training. Direct-based [48] further augments 3D-RCNN by adding mask pooling and offset flow. In contrast to these prior works, GSNet produces a more fine-grained 3D shape representation of vehicles by effective four-way feature fusion and *divide-and-conquer* shape reconstruction, which further inspires a geometrical scene aware loss to regularize network training with rich supervisory signals.

3 Pose and Shape Representation

6DoF Pose. The 6DoF pose for each instance consists of the 3D translation $\mathbf{T}$ and 3D rotation $\mathbf{R}$. $\mathbf{T}$ is represented by the object center coordinate $C_{obj} = \{x, y, z\}$ in the camera coordinate system C_{cam}. Rotation $\mathbf{R}$ defines the rotation Euler angles about X, Y, Z axes of the object coordinate system C_{obj}.

Divide-and-Conquer Shape Representation. We represent vehicle shape with dense mesh consisting of 1352 vertices and 2700 faces, which is much more fine-grained compared to the volume representation used in [22]. We start with

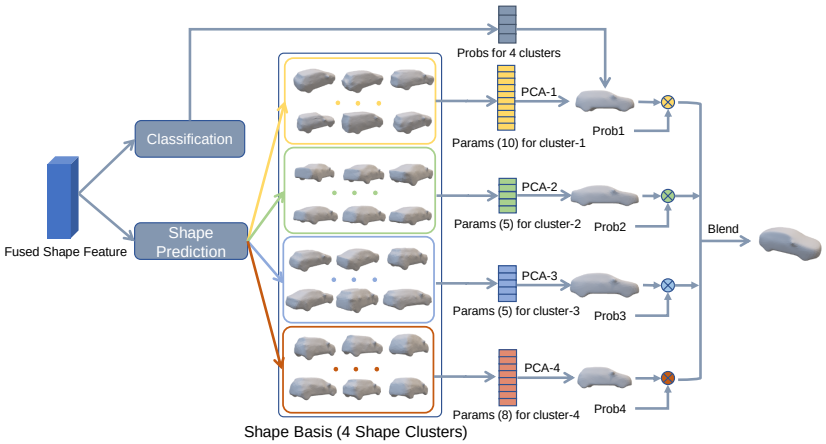

Fig. 2. Illustration of our divide-and-conquer 3D shape reconstruction module, where we obtain four independent PCA models for each shape cluster. Instance shape reconstruction is achieved by reconstructing shape in each cluster and blend them with the respective classification probabilities. This strategy achieves lower shape reconstruction error compared to other methods as shown in Table 4.

the CAD meshes provided by the ApolloCar3D database [48], which has different topology and vertex number for each car type. We convert them into the same topology with a fixed number of vertices by deforming a sphere using the SoftRas [34] method.

To ease the training of neural network for shape reconstruction, we reduce the shape representation dimension with principle component analysis (PCA) [41]. However, applying PCA to all available meshes directly [9,24,23] is sub-optimal due to the large variation of car types and shapes. We thus adopt a *divide-and-conquer* strategy as shown in Figure 2. We first cluster a total of 79 CAD models into four subsets with K-Means algorithm utilizing the shape similarity between car meshes. For each subset, we separately learn a low dimensional shape basis with PCA. Denote a subset of k vehicle meshes as $M = \{m_1, m_2, ..., m_k\}$, we use PCA to find $n \leq 10$ dimensional shape basis, $\bar{\mathbf{S}} \in \mathbb{R}^{N \times n}$, where $N \gg n$. During inference, the network classifies the input instance into the 4 clusters and predicts the principle component coefficient for each cluster. The final shape is blended from the four meshes weighted by the classification score. With this strategy, we achieve lower shape reconstruction error than directly applying PCA to all meshes or retrieval which is detailed in our ablation study.

4 Network Architecture Design

Figure 3 shows the overall architecture of our GSNet for joint car pose and shape reconstruction. We design and extract four types of features from a complex

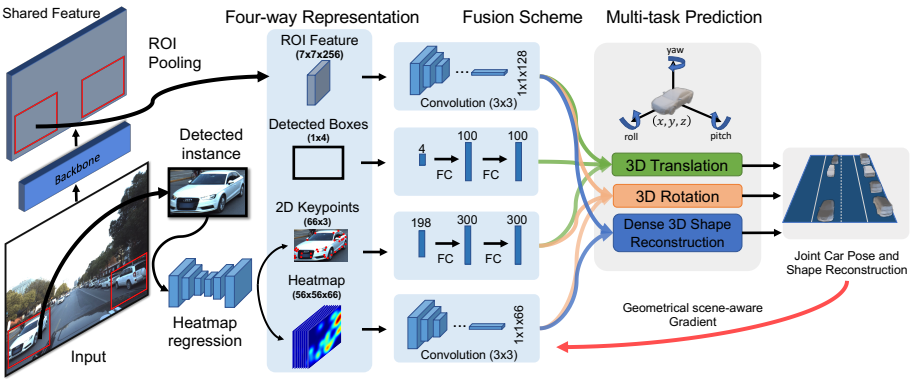

Fig. 3. Overview of our GSNet for joint vehicle pose and shape reconstruction. We use region-based 2D object detector [12] and a built-in heatmap regression branch to obtain ROI features, detected boxes, keypoint coordinates (global locations in the whole image) and corresponding heatmap (local positions and visibility in sub-region). GSNet performs an effective fusion of four-way input representations and builds three parallel branches respectively for 3D translation, rotation and shape estimation. 3D shape reconstruction is detailed in Figure 2 and our hybrid loss function is illustrated in section 5.

traffic scene, after which a fusion scheme is proposed to aggregate them. Finally, multi-task prediction is done in parallel to estimate 3D translation, rotation and shape via the intermediate fused representations.

Diverse Feature Extraction and Representation. Existing methods [22,59] only use ROI features to regress 3D parameters, but we argue that using diverse features can better extract useful information in a complex traffic scene. Given an input image, we first use a region-based 2D object detector [12] to detect car instances and obtain its global location. Based on the bounding boxes, ROI pooling is used to extract appearance features for each instance. In a parallel branch, each detected instance is fed to a fully-convolutional sub-network to obtain 2D keypoint heatmaps and coordinates. The coordinates encode rich geometric information that can hardly be obtained with ROI features alone [65], while the heatmaps encode part visibility to help the network discriminate occluded instances.

Detected boxes are represented as 2D box center (b_x, b_y), width b_w and height b_h in pixel space. Camera intrinsic calibration matrix is $[f_x, 0, p_x; 0, f_y, p_y; 0, 0, 1]$ where f_x, f_y are focal lengths in pixel units and (p_x, p_y) is the principal point at the image center. We transform b_x, b_y, b_w, b_h from pixel space to the corresponding coordinates u_x, u_y, u_w, u_h in the world frame:

$$u_x = \frac{(b_x - p_x)z}{f_x}, u_y = \frac{(b_y - p_y)z}{f_y}, u_w = \frac{b_w}{f_x}, u_h = \frac{b_h}{f_y}, \tag{1}$$

where z is the fixed scale factor. For keypoint localization, we use the 66 semantic keypoints for cars defined in [48]. A 2D keypoint is represented as $\mathbf{p_k} =$

$\{x_k, y_k, v_k\}$, where $\{x_k, y_k\}$ are the image coordinates and v_k denotes visibility. In implementation, we adapt [12] pre-trained for human pose estimation on COCO to initialize the keypoint localization branch. For extracting ROI features, we use FPN [31] as our backbone.

Fusion Scheme. We convert the extracted four-way inputs into 1D representation separately and decide which features to use for completing each task by prior knowledge. For global keypoint positions and detected boxes, we apply two fully-connected layers to convert them into higher level feature. For ROI feature maps and heatmaps, we adopt sequential convolutional operations with stride 2 to reduce their spatial size to 1×1 while keeping the channel number unchanged.

Instead of blindly using all features for prediction, we fuse different feature types that are most informative for each prediction branch. The translation $\mathbf{T}$ mainly affects the object location and scale during the imaging process, thus we concatenate the ROI feature, 2D keypoint feature and box position feature for translation regression. The rotation $\mathbf{R}$ determines the image appearance of the object given its 3D shape and texture, thus we utilize the fusion of ROI feature, heatmap feature and the keypoint feature as input. For estimating shape parameters $\mathbf{S}$, we aggregate the ROI and heatmap features.

Multi-task Prediction We design three *parallel* estimation branches (translation, rotation and shape reconstruction) as shown in Figure 3 since they are independent, where each branch directly regresses the targets with mutual benefits. Note that parts of input features such as ROI heatmap and keypoint positions are shared in different branches, which can be jointly optimized and is beneficial as shown by our experiments. In contrast to previous methods that predict translation or depth using a discretization policy [10], GSNet can directly regress the translation vector and achieve accurate result *without any post processing* or further refinement. For the shape reconstruction branch, the network classify the input instance and estimates the low-dimensional parameters (less than 30) for four clusters as described in section 3.

5 Geometrical and Scene-aware Supervision

To provide GSNet with rich supervisory signals, we design a composite loss functions consisting of multiple terms. Apart from ordinary regression losses, it also strives for geometrical consistency and considers scene-level constraints in both inter- and intra-instance manners.

Achieve geometrical consistency by projecting keypoints. With the rich geometric details of the 3D vehicles as shown in Figure 4, we exploit the 2D-3D keypoints correspondence using a pinhole camera model to provide extra supervision signal. For a 3D semantic keypoint $\mathbf{p}_k = (x_0, y_0, z_0)$ on the predicted mesh with translation $\mathbf{T}_{pred}$ and rotation $\mathbf{R}_{pred}$, the reprojection equation is:

$$P_{repro} = s \begin{bmatrix} u_0 \\ v_0 \\ 1 \end{bmatrix} = \mathbf{k}[\mathbf{R}_{pred}|\mathbf{T}_{pred}]\mathbf{p}_k, \tag{2}$$

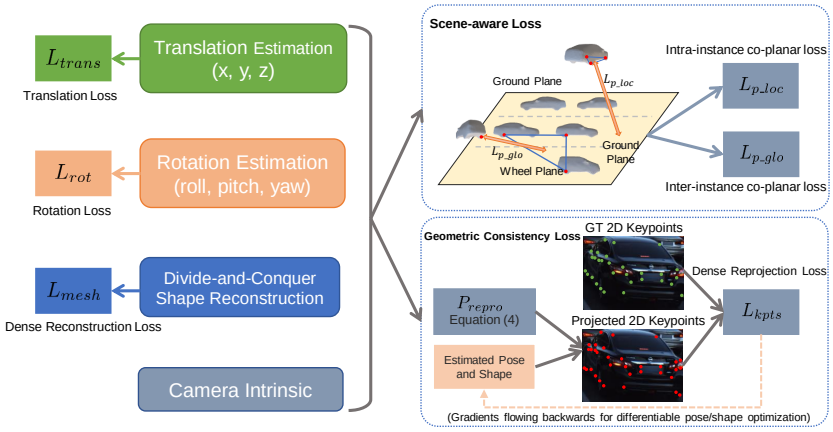

Fig. 4. The hybrid loss function for optimizing the GSNet. The scene-aware loss consists of two parts, L_{p_glo} for multiple car instances resting on common ground and L_{p_loc} for each single car at *a fine-grained level*. For geometrical consistency, camera intrinsics are used to project the predicted 3D semantic vertices on a car mesh to image and compared with the 2D detections.

where $\mathbf{k}$ is the camera intrinsic matrix and (u_0, v_0) is the projection point in pixels units. For the ith projected keypoint $\mathbf{p}_i = (u_i, v_i)$, the reprojection loss is

$$L_{kpt\_i} = \|\mathbf{p}_i - \bar{\mathbf{p}}_i\|_2^2, \tag{3}$$

where $\bar{\mathbf{p}}_i = (\bar{u}_i, \bar{v}_i)$ is the corresponding image evidence given by our heatmap regression module. The total loss L_{kpts} for n semantic keypoints in a car instance is

$$L_{kpts} = \sum_{i=1}^{n} L_{kpt\_i} V_i, \tag{4}$$

where V_i is a boolean value indicating the visibility of ith keypoint in the image. This reprojection loss is *differentiable* and can be easily incorporated in the end-to-end training process. The correspondence of 2D keypoints and 3D mesh vertices is needed to compute the loss and we determine it by ourselves. We project each 3D vertex on the ground truth mesh to image plane and find its nearest neighboring 2D points. The 66 3D vertices whose 2D projections have the most 2D annotated neighbors are selected as the corresponding 3D landmarks. We also provide an ablation experiment on the influence of keypoints number in the supplementary file.

Scene-aware Loss. Observe that most of cars rest on a common ground plane and the height of different instances is similar, thus the car centers are nearly co-planar. For each image, we locate mesh centers for four randomly-selected instances. Three of the centers define a plane $ax + by + cz + d = 0$ and denote the remaining car center coordinate as (x_1, y_1, z_1). As shown in Figure 4, we

introduce the *inter-instance co-planar loss* L_{p_glo} for multiple cars as:

$$L_{p\_glo} = \frac{|ax_1 + by_1 + cz_1 + d|}{\sqrt{a^2 + b^2 + c^2}}, \tag{5}$$

In addition, the centroids of the four wheels on a car should also lie in the same plane parallel to the ground. Thanks to the *dense* 3D mesh reconstructed by our multi-task network, we can readily obtain these four 3D coordinates. We thus propose the *intra-instance co-planar loss* L_{p_loc} to supplements L_{p_glo}. It is similar to Eq. 5 but the three points are chosen on the same instance.

Regression Losses. We use L2 loss L_{mesh} to penalize inaccurate 3D shape reconstruction as:

$$L_{mesh} = \frac{\sum_{j=1}^{m} \left\| \mathbf{M}_j - \bar{\mathbf{M}}_j \right\|_2^2}{m}, \tag{6}$$

where m is total number of vertices, $\mathbf{M}_j$ is the jth predicted vertex and $\bar{\mathbf{M}}_j$ is the ground truth vertex. For regression of 6DoF pose, we find that L1 loss performs better than L2 loss.The loss for translation regression is

$$L_{trans} = \left| \mathbf{T}_{pred} - \mathbf{T}_{gt} \right|, \tag{7}$$

where $\mathbf{T}_{gt}$ and $\mathbf{T}_{pred}$ are ground-truth and predicted translation vector, respectively. For regressing rotation in Euler angles, we restrict the range around each axis $[-\pi, \pi]$. Since this is a unimodal task, we define the regression loss as

$$L_{rot} = \begin{cases} |\mathbf{R}_{pred} - \mathbf{R}_{gt}| & \text{if } |\mathbf{R}_{pred} - \mathbf{R}_{gt}| \leq \pi, \\ 2\pi - |\mathbf{R}_{pred} - \mathbf{R}_{gt}| & \text{if } |\mathbf{R}_{pred} - \mathbf{R}_{gt}| > \pi, \end{cases} \tag{8}$$

where $\mathbf{R}_{pred}$ and $\mathbf{R}_{gt}$ are the predicted and ground truth rotation vector.

Sub-type Classification Loss. We also classify the car instance into 34 sub-types (sedan, minivan, SUV, etc.) and denote the classification loss as L_{cls}.

Final Objective Function. The final loss function L for training our GSNet is defined as:

$$\begin{aligned} L = {} & \lambda_{loc} L_{p\_loc} + \lambda_{glo} L_{p\_glo} + \lambda_{kpts} L_{kpts} \\ & + \lambda_{mesh} L_{mesh} + \lambda_{trans} L_{trans} + \lambda_{rot} L_{rot} + \lambda_{cls} L_{cls} \end{aligned} \tag{9}$$

where λs balance the above loss components. As validated by our experiments in section 6.2, this hybrid loss function design significantly promotes the network's performance compared to using only regression losses alone.

6 Experiments

6.1 Datasets and Experimental Settings

ApolloCar3D. We use the most recent and largest multi-task ApolloCar3D dataset [48] to train and evaluate GSNet. This dataset contains 5,277 high-resolution (2,710×3384) images. We follow the official split where 4036 images

are used for training, 200 for validation and the remaining 1041 for testing. Compared to KITTI [11], the instance count in ApolloCar3D is **20X** larger with far more cars per image (11.7 vs 4.8) where distant instances over 50 meters are also annotated. In addition, ApolloCar3D provides 3D shape ground truth to evaluate shape reconstruction quantitatively, which is not available in KITTI.

Pascal3D+ We also train and evaluate GSNet on Pascal3D+ [58] dataset using its car category. There are totally 6704 in-the-wild images with 1.19 cars per image on average. It also provides both dense 3D shape and 6D pose annotation.

Evaluation Metrics. We follow the evaluation metrics in [48], which utilizes instance 3D average precision (A3DP) with 10 thresholds (criteria from loose to strict) for **jointly** measuring translation, rotation and 3D car shape reconstruction accuracy. The results on the loose and strict criterion are respectively denoted as *c-l* and *c-s*. During evaluation, Euclidean distance is used for 3D translation while arccos distance is used for 3D rotation. For 3D shape reconstruction, a predicted mesh is rendered into 100 views to compute IoU with the ground truth masks and the mean IoU is used. In addition to the absolute distance error, the relative error in translation is also evaluated to emphasize the model performance for nearby cars, which are more important for autonomous driving. We denote A3DP evaluated in relative and absolute version as *A3DP-Rel* and *A3DP-Abs* respectively.

Implementation Details. GSNet utilizes the Mask R-CNN [12] with ResNet-101 backbone pre-trained on the COCO 2017 dataset [32] for object detection and extracting ROI features (7×7). We discard detected objects with confidence score less than 0.3. The $\lambda_{loc}, \lambda_{glo}, \lambda_{kpts}, \lambda_{mesh}, \lambda_{trans}, \lambda_{rot}, \lambda_{cls}$ in Eq. 9 are set to 5.0, 5.0, 0.01, 10.0, 0.5, 1.0, 0.5 to balance the loss components. During training, we use Adam optimizer [17] with initial learning rate 0.0025 and reduce it by half every 10 epochs for total 30 epochs. The 2D keypoint localization branch is trained separately where we use 4,036 training images containing 40,000 labeled vehicles with 2D keypoints and set threshold 0.1 for deciding keypoint visibility. When building the dense shape representation, there are respectively 9, 24, 14, 32 meshes in the four clusters.

6.2 Ablation Study of Network Architecture and Loss Design

We conduct three ablation experiments on ApolloCar3D validation set to validate our network design, loss functions and dense shape representation strategy.

Is extracting more features beneficial? We validate our four-way feature extraction fusion design by varying the number of used branches as: 1) Baseline: only using instance ROI features; 2) fusing transformed bounding box feature with the ROI feature; 3) combining predicted heatmap feature to the input; 4) further adding the 2D keypoint feature. The quantitative comparison is shown in Table 1. Compared to using ROI features alone, the injection of transformed detected boxes (center position, width and height) help provide geometric information, which help reduce translation error by 35.2% while improves *Rel-mAP* from 6.8 to 12.5 and *Abs-mAP* from 7.0 to 11.4. The introduction of keypoint

Table 1. Ablation study for GSNet on four-way feature fusion, which shows the relevant contribution of each representation with only regression losses. Performance is evaluated in terms of A3DP (*jointly* measuring translation, rotation and 3D car shape reconstruction accuracy), where *c-l* indicates results on loose criterion and *c-s* indicates strict criterion. GSNet exhibits a significant improvement compared to the baseline (with only ROI features), which promotes *A3DP-Rel* item *c-s* from 3.2 to 10.5. T and R in *6DoF Error* respectively represent 3D translation and rotation.

2D Input Representation				A3DP-Rel			A3DP-Abs			6DoF Error	
ROI	boxes	heatmap	kpts	mean	c-l	c-s	mean	c-l	c-s	T	R
✓				6.8	20.1	3.2	7.0	17.7	5.1	2.41	0.33
✓	✓			$12.5_{\uparrow 5.7}$	$30.1_{\uparrow 10.4}$	$8.9_{\uparrow 5.7}$	$11.4_{\uparrow 4.4}$	$26.6_{\uparrow 8.9}$	$8.8_{\uparrow 3.7}$	$1.56_{\downarrow 0.85}$	$0.32_{\downarrow 0.01}$
✓	✓	✓		$13.7_{\uparrow 6.9}$	$32.5_{\uparrow 12.4}$	$9.2_{\uparrow 6.0}$	$12.4_{\uparrow 5.4}$	$29.2_{\uparrow 11.5}$	$9.2_{\uparrow 4.1}$	$1.53_{\downarrow 0.88}$	$0.24_{\downarrow 0.09}$
✓	✓	✓	✓	$\mathbf{14.1}_{\uparrow 7.3}$	$\mathbf{32.9}_{\uparrow 12.8}$	$\mathbf{10.5}_{\uparrow 7.3}$	$\mathbf{12.8}_{\uparrow 5.8}$	$\mathbf{29.3}_{\uparrow 11.6}$	$\mathbf{9.9}_{\uparrow 4.8}$	$\mathbf{1.50}_{\downarrow 0.91}$	$\mathbf{0.24}_{\downarrow 0.09}$

heatmaps is beneficial especially for rotation estimation. This extra visibility information for the 2D keypoints reduces rotation error by 25.0% and further promoting *Rel-mAP* from 12.5 to 13.7 and *Abs-mAP* from 11.4 to 12.4. Finally, the 2D keypoint position branch complements the other three branches and improves model performance consistently for different evaluation metrics.

Effectiveness of the hybrid loss function. Here we fix our network architecture while varying the components of loss function to validate our loss design. The experiments are designed as follows: 1) Baseline: adopt four-way feature fusion architecture, but only train the network with regression and classification losses without shape reconstruction; 2) adding 3D shape reconstruction loss; 3) incorporating geometrical consistency loss; 4) adding scene-aware loss but only use the inter-instance version; 5) adding the intra-instance scene-aware component to complete the multi-task loss function. As shown in Table 2, the reprojection consistency loss promotes 3D localization performance significantly, where the 3D translation error reduces over 10% and *Rel-mAp* increases from 15.1 to 17.6. The scene-aware loss brings obvious improvement compared to ignoring the traffic scene context, especially for the *A3DP-Rel* strict criterion *c-s* (increasing AP from 14.2 to 19.8). In addition, using both inter-instance and intra-instance loss components outperforms using inter-instance scene-aware loss alone. Compared to the baseline, our hybrid loss function significantly promotes the performance of *Rel-mAp* and *Abs-mAp* respectively to 20.2 and 18.9.

Is jointly performing both tasks helpful? We argue that jointly performing dense shape reconstruction can in turn help 6D pose estimation. Without the introduction of the dense shape reconstruction task, we do not have access to the reconstruction loss C1 as well as the geometrical and scene-aware losses (C2, C3 and C4). Note that C1-C4 significantly improves estimation accuracy for translation and rotation.

Effectiveness of the divide-and-conquer strategy. Table 4 compares model performance using different shape representations: retrieval, single PCA shape-space model and our divide-and-conquer strategy detailed in section 3. Observe that our divide-and-conquer strategy not only reduces shape reconstruction error for around 10%, but also boosts the overall performance for traffic instance

Table 2. Ablation study for GSNet using different loss components of the hybrid loss function, which shows the relevant contribution of each component. C0, C1, C2, C3, C4 respectively denote pose regression loss, 3D shape reconstruction loss, geometrical consistency loss, inter-instance scene-aware loss and intra-instance scene-aware loss. GSNet exhibits a significant improvement compared to the baseline (with only regression losses), especially in estimating the surrounding car instances as shown by *A3DP-Rel* (item *c-s* has been significantly boosted from 10.5 to 19.8).

Loss Components					A3DP-Rel			A3DP-Abs			6DoF Error	
C0	C1	C2	C3	C4	mean	c-l	c-s	mean	c-l	c-s	T	R
✓					14.1	32.9	10.5	12.8	29.3	9.9	1.50	0.24
✓	✓				$15.1_{\uparrow 1.0}$	$34.8_{\uparrow 1.9}$	$11.3_{\uparrow 0.8}$	$15.0_{\uparrow 2.2}$	$32.0_{\uparrow 2.7}$	$13.0_{\uparrow 3.1}$	$1.44_{\downarrow 0.06}$	$0.23_{\downarrow 0.01}$
✓	✓	✓			$17.6_{\uparrow 3.5}$	$37.3_{\uparrow 4.4}$	$14.2_{\uparrow 3.7}$	$16.7_{\uparrow 3.9}$	$34.1_{\uparrow 4.8}$	$15.4_{\uparrow 5.5}$	$1.30_{\downarrow 0.20}$	$0.20_{\downarrow 0.04}$
✓	✓	✓	✓		$18.8_{\uparrow 4.7}$	$39.0_{\uparrow 6.1}$	$16.3_{\uparrow 5.8}$	$17.6_{\uparrow 4.8}$	$35.3_{\uparrow 6.0}$	$16.7_{\uparrow 6.8}$	$1.27_{\downarrow 0.23}$	$0.20_{\downarrow 0.04}$
✓	✓	✓	✓	✓	$\mathbf{20.2_{\uparrow 6.1}}$	$\mathbf{40.5_{\uparrow 7.6}}$	$\mathbf{19.8_{\uparrow 9.3}}$	$\mathbf{18.9_{\uparrow 6.1}}$	$\mathbf{37.4_{\uparrow 8.1}}$	$\mathbf{18.3_{\uparrow 8.4}}$	$\mathbf{1.23_{\downarrow 0.27}}$	$\mathbf{0.18_{\downarrow 0.06}}$

understanding. Also, we present shape reconstruction error distribution across different vehicle categories in our supplementary file.

6.3 Comparison with state-of-the-art methods

Quantitative Comparison on ApolloCar3D We compare GSNet with state-of-the-art approaches that jointly reconstruct vehicle pose and shape on ApolloCar3D dataset as shown in Table 3. The most recent *regression-based* approaches are: 1) 3D-RCNN [22], which regress 3D instances from ROI features and add geometrical consistency by designing a differentiable render-and-compare mask loss; 2) Direct-based method in [48], which improves 3D-RCNN by adding mask pooling and offset flow. We can see that our GSNet achieves superior results among the existing *regression-based* methods across the evaluation metrics while being fast, nearly doubling the mAP performance of 3D-RCNN in *A3DP-Rel* entry. Compared to the *fitting-based* pose estimation methods using Epnp [25], which fit 3D template car model to 2D image observations in a time-consuming optimization process, GSNet performs comparably in *A3DP-Rel* and *A3DP-Abs* metrics with a high-resolution shape reconstruction output not constrained by the existing CAD templates. Note that *fitting-based* methods consume long time and thus are not feasible for time-critical applications. Also note that *A3DP-Rel* is important since nearby cars are more relevant for self-driving car to make motion planning, where GSNet improves *c-l* AP performance by 15.75 compared to Kpts-based [48].

Quantitative Comparison on Pascal3D+ To further validate our network, we evaluate GSNet on the Pascal3D+ [58] dataset using its car category. We follow the setting in [22,37] to evaluate the viewpoint and use $Acc_{\pi/6}$ and $MedErr$ adopted in [37,53] to report results in Table 5, where the median angular error improves by 20% from 3.0° to 2.4° compared to 3D-RCNN.

Qualitative Analysis Figure 5 shows qualitative comparisons with other direct *regression-based* methods for joint vehicle pose and shape reconstruction.

Table 3. Performance comparison with state-of-the-art 3D joint vehicle pose and shape reconstruction algorithms on ApolloCar3D dataset. Times is the average inference time for processing each image. GSNet achieves significantly better performance than state-of-the-art regression-based approaches (using a deep network to directly estimate the pose/shape from pixels) with both high precision and fast speed where inference time is *critical* in autonomous driving. * denotes fitting-based methods, which fits a 3D template car model to best match its 2D image observations (requires no image with 3D ground truth) and is time-consuming.

Model	Shape Reconstruction	Regression-based	A3DP-Rel			A3DP-Abs			Times	Time-efficient
			mean	c-l	c-s	mean	c-l	c-s		
DeepMANTA (CVPR'17) [3]*	retrieval	✗	16.04	23.76	19.80	20.10	30.69	23.76	3.38s	✗
Kpts-based (CVPR'19) [48]*	retrieval	✗	16.53	24.75	19.80	20.40	31.68	24.75	8.5s	✗
3D-RCNN (CVPR'18) [22]	TSDF volume [8]	✓	10.79	17.82	11.88	16.44	29.70	**19.80**	0.29s	✓
Direct-based (CVPR'19) [48]	retrieval	✓	11.49	17.82	11.88	15.15	28.71	17.82	0.34s	✓
Ours: GSNet	Detailed deformable mesh	✓	**20.21**	**40.50**	**19.85**	**18.91**	**37.42**	18.36	0.45s	✓

Table 4. Results comparison between GSNet adopting retrieval, single PCA model and our divide-and-conquer shape module on ApolloCar3D validation set.

Shape-space Model	Shape Reconstruction Error	Rel-mAP
Retrieval	92.46	17.6
Single PCA	88.68	18.7
Divide-and-Conquer Shape Module	**81.33**	**20.2**

Table 5. Results on viewpoint estimation with annotated boxes on Pascal3D+ [58] for *Car*, where GSNet gets highest accuracy and lowest angular error.

Model	$Acc_{\pi/6} \uparrow$	$MedErr \downarrow$
RenderForCNN [49]	0.88	6.0°
Deep3DBox [37]	0.90	5.8°
3D-RCNN [22]	0.96	3.0°
Ours: GSNet	**0.98**	**2.4°**

Fig. 5. Qualitative comparison on the ApolloCar3D test set of different approaches by rendering 3D mesh output projected onto the input 2D image. The first row are the input images, the second row is the result of Direct-based [48] and the third row is predicted by our GSNet. The bottom row shows the reconstructed meshes in 3D space. Corresponding car instances are depicted in the same color.

Compared with Direct-based [48], our GSNet produces more accurate 6DoF pose estimation and 3D shape reconstruction from monocular images due to the

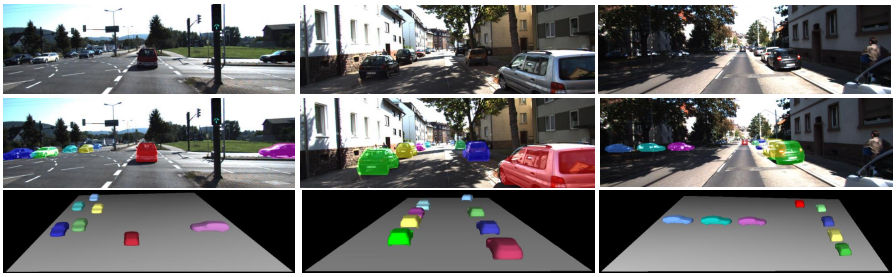

Fig. 6. Cross-dataset generalization of GSNet on KITTI [11] dataset. The first row are the input images and the second row are our reconstructed 3D car meshes projected onto the original image. Additional results are shown in our supplementary material.

effective four-way feature fusion, the hybrid loss which considers both geometrical consistency and scene-level constraints and our divide-and-conquer shape reconstruction. Although directly regressing depth based on monocular images is considered as an ill-posed problem, our GSNet achieves high 3D estimation accuracy (our projected masks of car meshes on input images show an almost perfect match), particularly for instances in close proximity to the self-driving vehicle. The last column of the figure shows that the estimation of GSNet is still robust even in a relatively dark environment where the two left cars are heavily occluded. The last row visualizes the predicted 3D vehicle instances.

Figure 6 shows additional qualitative results on applying GSNet on KITTI [11]. Despite that GSNet is not trained on KITTI, the generalization ability of our model is validated as can be seen from the accurate 6D pose estimation and shape reconstruction of unseen vehicles. More results (including 3 temporally preceding frames of KITTI) are available in our supplementary material.

7 Conclusion

We present an end-to-end multi-task network GSNet, which jointly reconstructs 6DoF pose and 3D shape of vehicles from single urban street view. Compared to previous regression-based methods, GSNet not only explores more potential feature sources and uses an effective fusion scheme to supplement ROI features, but also provides richer supervisory signals from both geometric and scene-level perspectives. Vehicle pose estimation and shape reconstruction are tightly integrated in our system and benefit from each other, where 3D reconstruction delivers geometric scene context and greatly helps improve pose estimation precision. Extensive experiments conducted on ApolloCar3D and Pascal3D+ have demonstrated our state-of-the-art performance and validated the effectiveness of GSNet with both high accuracy and fast speed.

Acknowledgement: This research is supported in part by the Research Grant Council of the Hong Kong SAR under grant no. 1620818.

References

1. Brazil, G., Liu, X.: M3d-rpn: Monocular 3d region proposal network for object detection. In: ICCV (2019)
2. Cao, Z., Sheikh, Y., Banerjee, N.K.: Real-time scalable 6dof pose estimation for textureless objects. In: 2016 IEEE International conference on Robotics and Automation (ICRA) (2016)
3. Chabot, F., Chaouch, M., Rabarisoa, J., Teulière, C., Chateau, T.: Deep manta: A coarse-to-fine many-task network for joint 2d and 3d vehicle analysis from monocular image. In: CVPR (2017)
4. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012 (2015)
5. Chen, X., Kundu, K., Zhang, Z., Ma, H., Fidler, S., Urtasun, R.: Monocular 3d object detection for autonomous driving. In: CVPR (2016)
6. Chen, X., Ma, H., Wan, J., Li, B., Xia, T.: Multi-view 3d object detection network for autonomous driving. In: CVPR (2017)
7. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: CVPR (2016)
8. Curless, B., Levoy, M.: A volumetric method for building complex models from range images. In: SIGGRAPH (1996)
9. Engelmann, F., Stückler, J., Leibe, B.: Samp: shape and motion priors for 4d vehicle reconstruction. In: WACV (2017)
10. Fu, H., Gong, M., Wang, C., Batmanghelich, K., Tao, D.: Deep ordinal regression network for monocular depth estimation. In: CVPR (2018)
11. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: CVPR (2012)
12. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: ICCV (2017)
13. Hinterstoisser, S., Cagniart, C., Ilic, S., Sturm, P., Navab, N., Fua, P., Lepetit, V.: Gradient response maps for real-time detection of textureless objects. TPAMI **34**(5), 876–888 (2011)
14. Hu, Y., Hugonot, J., Fua, P., Salzmann, M.: Segmentation-driven 6d object pose estimation. In: CVPR (2019)
15. Kar, A., Tulsiani, S., Carreira, J., Malik, J.: Category-specific object reconstruction from a single image. In: CVPR (2015)
16. Kehl, W., Manhardt, F., Tombari, F., Ilic, S., Navab, N.: Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In: ICCV (2017)
17. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
18. Kolotouros, N., Pavlakos, G., Black, M.J., Daniilidis, K.: Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In: ICCV (2019)
19. Kong, C., Lin, C.H., Lucey, S.: Using locally corresponding cad models for dense 3d reconstructions from a single image. In: CVPR (2017)
20. Krishna Murthy, J., Sai Krishna, G., Chhaya, F., Madhava Krishna, K.: Reconstructing vehicles from a single image: Shape priors for road scene understanding. In: 2017 IEEE International Conference on Robotics and Automation (ICRA) (2017)
21. Ku, J., Pon, A.D., Waslander, S.L.: Monocular 3d object detection leveraging accurate proposals and shape reconstruction. In: CVPR (2019)

22. Kundu, A., Li, Y., Rehg, J.M.: 3d-rcnn: Instance-level 3d object reconstruction via render-and-compare. In: CVPR (2018)
23. Leotta, M.J., Mundy, J.L.: Predicting high resolution image edges with a generic, adaptive, 3-d vehicle model. In: CVPR (2009)
24. Leotta, M.J., Mundy, J.L.: Vehicle surveillance with a generic, adaptive, 3d vehicle model. TPAMI **33**(7), 1457–1469 (2010)
25. Lepetit, V., Moreno-Noguer, F., Fua, P.: Epnp: An accurate o(n) solution to the pnp problem. IJCV **81**(2), 155 (2009)
26. Li, C., Zeeshan Zia, M., Tran, Q.H., Yu, X., Hager, G.D., Chandraker, M.: Deep supervision with shape concepts for occlusion-aware 3d object parsing. In: CVPR (2017)
27. Li, P., Chen, X., Shen, S.: Stereo r-cnn based 3d object detection for autonomous driving. In: CVPR (2019)
28. Li, P., Qin, T., Shen, S.: Stereo vision-based semantic 3d object and ego-motion tracking for autonomous driving. In: ECCV (2018)
29. Liang, M., Yang, B., Wang, S., Urtasun, R.: Deep continuous fusion for multi-sensor 3d object detection. In: ECCV (2018)
30. Lin, C.H., Wang, O., Russell, B.C., Shechtman, E., Kim, V.G., Fisher, M., Lucey, S.: Photometric mesh optimization for video-aligned 3d object reconstruction. In: CVPR (2019)
31. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: CVPR (2017)
32. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014)
33. Liu, L., Lu, J., Xu, C., Tian, Q., Zhou, J.: Deep fitting degree scoring network for monocular 3d object detection. In: CVPR (2019)
34. Liu, S., Li, T., Chen, W., Li, H.: Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In: ICCV (2019)
35. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: ECCV (2016)
36. Mottaghi, R., Xiang, Y., Savarese, S.: A coarse-to-fine model for 3d pose estimation and sub-category recognition. In: CVPR (2015)
37. Mousavian, A., Anguelov, D., Flynn, J., Kosecka, J.: 3d bounding box estimation using deep learning and geometry. In: CVPR (2017)
38. Pavlakos, G., Zhou, X., Chan, A., Derpanis, K.G., Daniilidis, K.: 6-dof object pose from semantic keypoints. In: 2017 IEEE International Conference on Robotics and Automation (ICRA) (2017)
39. Peng, S., Liu, Y., Huang, Q., Zhou, X., Bao, H.: Pvnet: Pixel-wise voting network for 6dof pose estimation. In: CVPR (2019)
40. Pohlen, T., Hermans, A., Mathias, M., Leibe, B.: Full-resolution residual networks for semantic segmentation in street scenes. In: CVPR (2017)
41. Prisacariu, V.A., Reid, I.: Nonlinear shape manifolds as shape priors in level set segmentation and tracking. In: CVPR (2011)
42. Rad, M., Lepetit, V.: Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In: ICCV (2017)
43. Richter, S.R., Roth, S.: Matryoshka networks: Predicting 3d geometry via nested shape layers. In: CVPR (2018)
44. Riegler, G., Osman Ulusoy, A., Geiger, A.: Octnet: Learning deep 3d representations at high resolutions. In: CVPR (2017)

45. Rothganger, F., Lazebnik, S., Schmid, C., Ponce, J.: 3d object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints. IJCV **66**(3), 231–259 (2006)
46. Simonelli, A., Bulò, S.R.R., Porzi, L., López-Antequera, M., Kontschieder, P.: Disentangling monocular 3d object detection. In: ICCV (2019)
47. Sinha, A., Unmesh, A., Huang, Q., Ramani, K.: Surfnet: Generating 3d shape surfaces using deep residual networks. In: CVPR (2017)
48. Song, X., Wang, P., Zhou, D., Zhu, R., Guan, C., Dai, Y., Su, H., Li, H., Yang, R.: Apollocar3d: A large 3d car instance understanding benchmark for autonomous driving. In: CVPR (2019)
49. Su, H., Qi, C.R., Li, Y., Guibas, L.J.: Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views. In: ICCV (2015)
50. Sundermeyer, M., Marton, Z.C., Durner, M., Brucker, M., Triebel, R.: Implicit 3d orientation learning for 6d object detection from rgb images. In: ECCV (2018)
51. Tekin, B., Sinha, S.N., Fua, P.: Real-time seamless single shot 6d object pose prediction. In: CVPR (2018)
52. Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., Abbeel, P.: Domain randomization for transferring deep neural networks from simulation to the real world. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2017)
53. Tulsiani, S., Malik, J.: Viewpoints and keypoints. In: CVPR (2015)
54. Wagner, D., Reitmayr, G., Mulloni, A., Drummond, T., Schmalstieg, D.: Pose tracking from natural features on mobile phones. In: IEEE/ACM International Symposium on Mixed and Augmented Reality (2008)
55. Wu, J., Xue, T., Lim, J.J., Tian, Y., Tenenbaum, J.B., Torralba, A., Freeman, W.T.: Single image 3d interpreter network. In: ECCV (2016)
56. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3d shapenets: A deep representation for volumetric shapes. In: CVPR (2015)
57. Xiang, Y., Choi, W., Lin, Y., Savarese, S.: Data-driven 3d voxel patterns for object category recognition. In: CVPR (2015)
58. Xiang, Y., Mottaghi, R., Savarese, S.: Beyond pascal: A benchmark for 3d object detection in the wild. In: WACV (2014)
59. Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. Robotics: Science and Systems (RSS) (2018)
60. Xu, B., Chen, Z.: Multi-level fusion based 3d object detection from monocular images. In: CVPR (2018)
61. Yan, X., Yang, J., Yumer, E., Guo, Y., Lee, H.: Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In: NIPS (2016)
62. Yang, B., Luo, W., Urtasun, R.: Pixor: Real-time 3d object detection from point clouds. In: CVPR (2018)
63. Yang, Z., Sun, Y., Liu, S., Shen, X., Jia, J.: Std: Sparse-to-dense 3d object detector for point cloud. In: ICCV (2019)
64. Zeeshan Zia, M., Stark, M., Schindler, K.: Are cars just 3d boxes?-jointly estimating the 3d shape of multiple objects. In: CVPR (2014)
65. Zhao, R., Wang, Y., Martinez, A.M.: A simple, fast and highly-accurate algorithm to recover 3d shape from 2d landmarks on a single image. TPAMI **40**(12), 3059–3066 (2017)
66. Zhu, R., Kiani Galoogahi, H., Wang, C., Lucey, S.: Rethinking reprojection: Closing the loop for pose-aware shape reconstruction from a single image. In: ICCV (2017)

67. Zia, M.Z., Stark, M., Schiele, B., Schindler, K.: Detailed 3d representations for object recognition and modeling. TPAMI **35**(11), 2608–2623 (2013)