The Earth ain't Flat: Monocular Reconstruction of Vehicles on Steep and Graded Roads from a Moving Camera

Junaid Ahmed Ansari[1]*, Sarthak Sharma[1]*, Anshuman Majumdar[1],
J. Krishna Murthy[2] and K. Madhava Krishna[1]

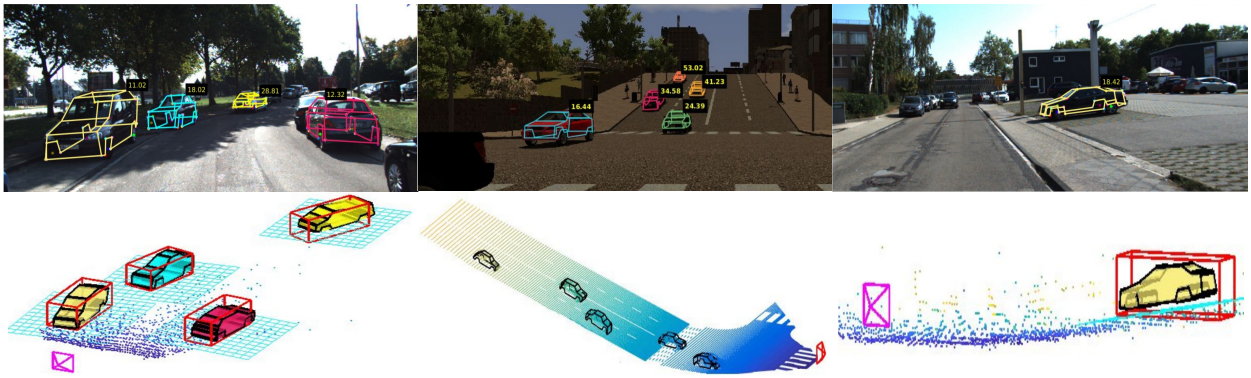

Fig. 1. Outputs of the proposed monocular object localization system. The system is capable of estimating the shape and pose (without scale-factor ambiguity) of objects located on surfaces that are non-coplanar with the moving ego vehicle. *Top*: Projection of the estimated shapes (wireframes) of cars. Above each car, distance of the car from the camera is shown (in meters). *Bottom*: Estimated wireframe and road points in 3D. For the first and third columns, estimated wireframes are compared with their respective ground truth 3D bounding boxes (in red), highlighting the accurate localization of the objects. In the second scene, we show the estimated cars in 3D, overlaid on a dense ground truth 3D point cloud consisting of road surface and the target vehicles. Notice how even objects over 50 meters away on steep slopes are accurately localized.

***Abstract*— Accurate localization of other traffic participants is a vital task in autonomous driving systems. State-of-the-art systems employ a combination of sensing modalities such as RGB cameras and LiDARs for localizing traffic participants, but monocular localization demonstrations have been confined to plain roads. We demonstrate — to the best of our knowledge — the first results for monocular object localization and shape estimation on surfaces that are non-coplanar with the moving ego vehicle mounted with a monocular camera. We approximate road surfaces by local planar patches and use semantic cues from vehicles in the scene to initialize a local bundle-adjustment like procedure that simultaneously estimates the 3D pose and shape of the vehicles, and the orientation of the local ground plane on which the vehicle stands. We also demonstrate that our approach transfers from synthetic to real data, without any hyperparameter-/fine-tuning. We evaluate the proposed approach on the KITTI and SYNTHIA-SF benchmarks, for a variety of road plane configurations. The proposed approach significantly improves the state-of-the-art for monocular object localization on arbitrarily-shaped roads.**

I. Introduction

With the advent and subsequent commercialization of autonomous driving, there has been an increased interest in monocular object localization for urban driving scenarios. While recent monocular localization methods [1], [2] achieve better localization precision when compared with stereo methods, they are confined to scenarios where the road is (very nearly) flat. Other monocular object localization systems [3], [4] too have such a limiting assumption.

Reconstructing vehicles from a monocular camera is a challenging task, owing to several factors, such as dearth of stable feature tracks on moving vehicles, self-occlusion, and is ill-posed if the camera itself is in motion. To overcome them, discriminative features [5] and shape priors [2], [6] have been used to pose a bundle adjustment like scheme [2] that solves for the shape and pose of a detected vehicle, assuming a prior on the shapes of all instances from a category. Use of shape priors results in a richer representation of reconstructed vehicles (3D wireframes rather than 3D bounding boxes).

We present — to the best of our knowledge — the first results for monocular object shape and pose estimation on surfaces that are non-coplanar with the moving ego vehicle. We approximate road surfaces by local planar patches and use semantic cues from vehicles in the scene to initialize a local bundle-adjustment like procedure that simultaneously estimates the pose and shape of the vehicles, and the orientation of the local ground plane on which it stands. Using the proposed approach, we accurately reconstruct vehicles, *predominantly using cues from a single image only*. This method works across a variety of road geometries and improves the vehicle localization accuracy on extremely steep and non-planar roads substantially.

To evaluate our approach, we use KITTI [7] and SYNTHIA-SF [8] benchmarks. While sequences from KITTI [7] dataset only have mild-to-moderate slopes and banks, it provides a fair comparison with other baselines [1], [2]. Whereas, SYNTHIA-SF [8] has extremely steep roads and

*The first two authors contributed equally to this work.

[1]Junaid Ahmed Ansari, Sarthak Sharma, Anshuman Majumdar, and K. Madhava Krishna are with the Robotics Research Center, KCIS, IIIT Hyderabad, India. junaid.ansari@research.iiit.ac.in

[2]J. Krishna Murthy is with Montreal Institute of Learning Algorithms (MILA), Universite de Montreal, Canada.

This work was supported by grants made available by Qualcomm Innovation Fellowship India, 2017.

demonstrates the efficacy of the proposed approach in adapting to a wide range of road surfaces.

II. Related Work

In this section, we briefly review the relevant literature and contrast it with our proposed approach.

A. Shape Priors

Shape priors have been widely used in [6], [9], [2] to ease the task of object reconstruction. The underlying hypothesis is that the shape of any instance from a category can be represented as a linear combination of the deformations of the category's mean shape along certain directions, called *basis vectors*. This linear subspace model was used to formulate a stochastic hill climbing problem in [6] to estimate the shape and pose of a vehicle in a single image. However, this is prohibitively slow to be used in real-time.

B. Monocular Localization in Urban Driving Scenarios

Estimating the 3D shape and pose from a single image has attracted a lot of interest in recent years, supported with the availability of datasets like KITTI [7] and ShapeNet [10].

Approaches like [1], [11] follow a 3D-2D pipeline that involves modeling the 3D shape offline and then solving for 3D deformations in it using localized 2D keypoints in RGB image as evidence, overcoming the need to explicitly estimate the 3D keypoints. [1] presents an approach to estimate the 3D shape and pose of the vehicles from a single image. The 3D shape of a vehicle is modeled using a shape prior based on a linear subspace model and deformation coefficients are estimated by solving an optimization problem with 2D keypoints, localized using a CNN.

In [3] and [12], the authors develop a real-time monocular SfM system using information from multiple image frames. However, vehicles are represented as 3D bounding boxes. It was demonstrated in [2] that having a richer representation for the vehicle (3D wireframe), significantly boosts localization accuracy. Mono3D [13] trains a CNN that jointly performs object detection in 2D and 3D space and estimates oriented bounding boxes for vehicles. Although it outperforms stereo competitors, it assumed planar road surfaces.

Similarly, [1], [2], [3], [12] rely on the coplanarity assumption for the localized vehicle and the ego car. Most of these methods use the approach outlined in [4] to estimate the depth of a vehicle under the coplanarity assumption.

C. Monocular Road Surface Reconstruction

There is relatively little work on road surface estimation from a monocular camera. In [14], the authors propose a simple road edge prediction framework using edges and lanes detected in earlier frames. No surface level reconstruction is provided. In [15], road width and shape of the drivable area are estimated using a Conditional Random Field (CRF).

In contrast to the above approaches, the proposed approach is independent of the road plane profile of vehicles and is capable of accurately localizing the vehicles. The method outperforms the current best competitor [2] by a significant margin, highlighting how the existing approaches fail to deal with vehicles on arbitrarily oriented road surfaces.

III. Geometry and Object Shape Costs

A. Background: Shape Priors

In this section, we outline our approach to reconstruct vehicles on arbitrarily oriented roads surfaces.

Along the lines of [6], [1], [2], we assume that each vehicle (in this case, a car) is represented in 3D by a wireframe consisting of K vertices (we use $K = 36$, according to the setup illustrated in Fig. 4), each of which has a unique semantic meaning. For instance, these vertices correspond to locations of headlights, tail lights, wheel centers, rooftop corners, etc. that are easily identifiable across all cars. We use a set of aligned 945 CAD models of cars from the ShapeNet [10] repository and annotate each of them with K keypoint locations in 3D. We then use the render pipeline presented in [16] to synthesize a dataset comprising about 1.2 million images of rendered cars with annotated 2D keypoint locations. Over this dataset, we train a keypoint localization network based on the stacked-hourglass architecture [5]. We use this CNN, trained entirely on synthetic data, across all experiments reported in this work. We observe that the network generalizes well to real data, consistent with the findings in [17].

Using notation from [2], we denote the mean wireframe for the vehicle category by $\bar{X} \in \mathbb{R}^{3K}$. The basis vectors are stacked into a $3K \times B$ matrix denoted V. The deformation coefficients (also referred to as the shape parameters) $\lambda \in \mathbb{R}^B$ uniquely determine the shape of a particular instance. If we assume that the object coordinate frame has a rotation $R \in SO(3)$ and translation $t \in \mathbb{R}^3$ with respect to the camera center, any instance X can then be parameterized by the shape prior model as (pictorially illustrated in Fig. 4)

$$X = \hat{R}\left(\bar{X} + V\lambda\right) + \hat{t} \tag{1}$$

Here, $\hat{R} = diag([R, R, ..., R]) \in \mathbb{R}^{3K \times 3K}$, and $\hat{t} = \left(t^T, t^T, ..., t^T\right)^T \in \mathbb{R}^{3K}$. $\bar{X} = \left(\bar{X}_1^T, \bar{X}_2^T, ..., \bar{X}_K^T\right)^T$ is an ordered collection of the 3D locations of the keypoints in the mean wireframe.

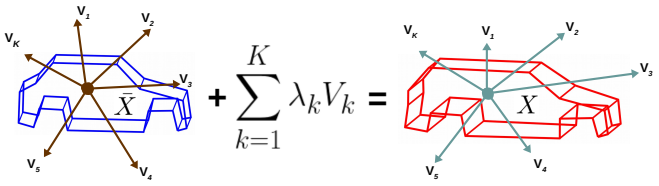

Fig. 4. Illustrating linear combination of deformations of a mean shape along its basis vectors to produce any other shape in the category

If we denote the locations of an ordered collection of 2D keypoints by $\hat{x} = \left(\hat{x}_1^T, \hat{x}_2^T, ..., \hat{x}_K^T\right)^T \in \mathbb{R}^{2K}$, the pose (R, t) and shape (λ) of the vehicle can be obtained by minimizing the following objective function in an alternating fashion - once for pose, and once for shape.

$$\min_{R,t,\lambda} \mathcal{L}_r = \|\pi_K\left(\hat{R}\left(\bar{X} + V\lambda\right) + \hat{t}; f_x, f_y, c_x, c_y\right) - \hat{x}\|_2^2 \tag{2}$$

$\pi_K()$ is a vectorized version of the perspective projection operator, which takes in K 3D points and computes

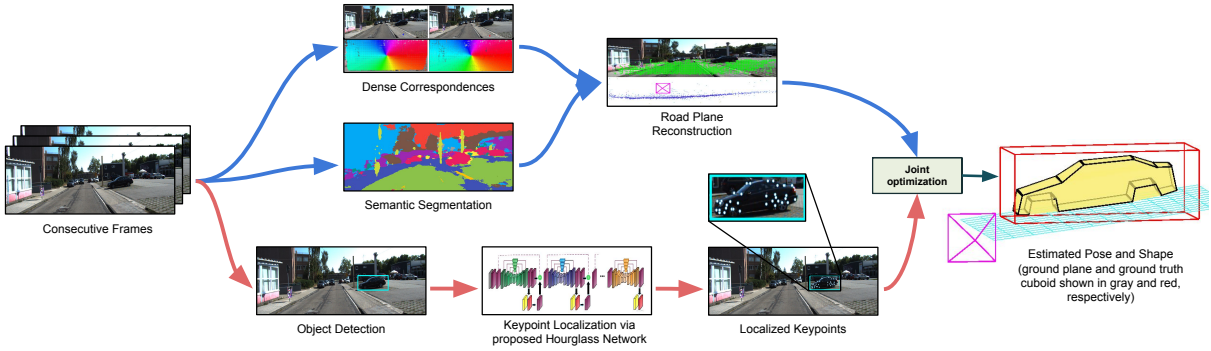

Fig. 3. Illustration of the proposed pipeline. The system takes 3 consecutive frames (in case of no lane markers). In the upper half (blue arrows), we illustrate the method for estimating the ground plane i.e. using dense correspondences over the frames and then performing bundle adjustment. In the lower half (red arrows), the detected bounding boxes in each frame are processed using the proposed keypoint localization CNN to obtain 2D locations of a discriminative set of semantic parts. The pose and shape of the object are then adjusted by incorporating the estimated ground plane information.

their image coordinates, given the camera intrinsics $\mu = (f_x, f_y, c_x, c_y)$. Specifically, π_K is the following function.

$$\pi\left((X,Y,Z)^T;\mu\right) = \begin{pmatrix} \frac{f_x X}{Z} + c_x \\ \frac{f_y Y}{Z} + c_y \end{pmatrix}$$
$$\pi_K\left((X_1^T,...,X_K^T)^T;\mu\right) = \left(\pi(X_1;\mu)^T,...,\pi(X_K;\mu)^T\right)^T \quad (3)$$

B. System Setup

We operate on image streams captured by a front-facing monocular (RGB) camera mounted on a car. The height H above the ground at which the camera is assumed to be known a priori (this helps in resolving scale-factor ambiguity in monocular reconstruction).

We assume that, on each incoming image, an object detector [18] runs and detects vehicles in the image (as bounding boxes). We also perform a semantic segmentation of the input image using the SegNet [19] convolutional architecture. The proposed pipeline is illustrated in Fig. 3.

C. Reconstruction of Vehicles on Slopes

To formulate a lightweight, yet robust optimization problem for reconstructing vehicles on non-planar road surfaces(i.e. roads with slopes and banks), we assume that the road is locally planar. By this, we mean that the patch of the road that lies exactly beneath a detected vehicle is assumed to be a planar patch. This assumption is corroborated by [3], where allowing each vehicle to have an adaptive local ground plane boosts localization accuracy.

Each detected vehicle v is on a planar patch parameterized by (n_g^{vT}, d_g^v), where n_g^v is a vector that denotes the normal to the planar patch and d_g^v denotes the distance of the planar patch from the origin of the camera coordinate frame.

D. Resolution of Scale-Factor Ambiguity

Monocular camera setups inherently suffer from scale-factor ambiguity, i.e., any 3D length estimated from a set of images is accurate up to a positive scalar. But, for the autonomous driving applications, we require that vehicles are localized in *metric scale*, i.e., in real-world units (such as meters, for instance). We resolve scale ambiguity using one of the following two approaches.

Using Dimensions of Detected Lanes: Most roads have lane marking or zebra crossings of standard dimensions that are known to us a priori. We use the method from [20] to detect lane markings, and if we know the height of the camera above the ground and the dimensions of the lane markings, we can retrieve the planar patch comprising the lane marking and the distance to that lane marking (in meters). Such a method estimates the local ground plane (of a lane marking near the vehicle) using information from just a single image.

Using 3-View Reconstruction and Camera Height: The above method can only be employed on roads where there are lane markings and in particular only if a lane marking is detected near a vehicle, which is not true for all scenarios we encounter. In the more general case, we can recover absolute (metric) scale by using the following 3-view reconstruction scheme. Assume we have three consecutive frames f_1, f_2, f_3 with sufficient parallax. We use DeepMatching[21] for establishing dense correspondences between frames f_1 to f_2. Then, using a sufficient mix of road and non-road points, we estimate the egomotion between the frames using standard multi-view motion estimation techniques [22]. Using the estimated egomotion, we triangulate points *close*[1] to the car that lie on the road surface and add points from frame f_3 to the reconstruction[2]. A local ground plane patch can then be estimated by estimating a dominant plane from the obtained point cloud using a RANSAC-like routine. Once such a plane is obtained, we can scale the reconstruction such that the median of the Y-coordinates of the estimated plane is roughly equal to the height of the camera above the ground (which is assumed to be known during initial setup).

[1]We expand the car bounding box by a factor of 1.9 to 2.0, and pick all points from the expanded bounding box that are classified as *road* by SegNet [19].

[2]This is typically done by propagating feature matches from frame f_2 to frame f_3, and running a resection routine to estimate the egomotion between frame f_1 and frame f_3, and then triangulating points from f_3 onto the initial reconstruction [22]

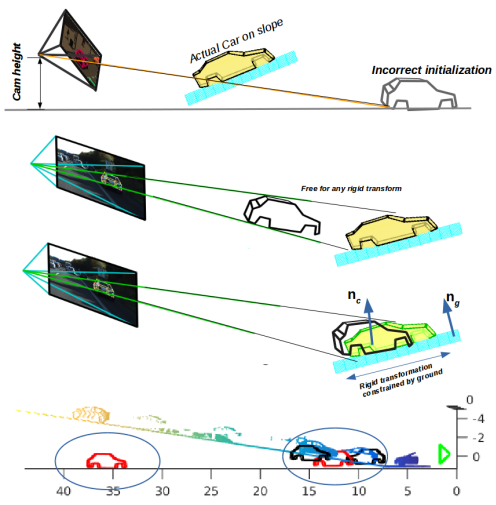

Fig. 5. **How does ground plane help?** From top to bottom - (i) Illustrating how coplanarity assumption results in incorrect initialization in existing approaches (ii) Relying only on minimizing the reprojection error, leaves the optimizer free to rigidly transform the mean car (iii) Joint optimization constrains the car to be on ground while minimizing the reprojection error, resulting in more accurate reconstruction and localization (n_c and n_g are car base and road plane normals respectively) (iv) Failure of coplanarity assumption for steep roads on SYNTHIA-SF [8]. Notice the incorrect initialization of the car on slopes via method proposed by [1], shown in red. Our method is not bound by this coplanarity assumption and initializes the vehicle correctly, shown in black. We overlay the initialized wireframe on the ground truth 3D points for comparison.

E. Joint Optimization for Ground Plane and Vehicle Pose and Shape Estimation

Equation 2 represents the optimization problem that is solved to estimate the shape and pose of a vehicle from just a single image or from a pair of images whenever available [2]. However, this formulation assumes coplanarity of the ego car and of the object being reconstructed. We illustrate in Fig. 5 that drastic errors in localization result when the assumption does not hold and how using the local ground plane circumvents this problem.

We assume that, in the current frame, a set of vehicles $\mathcal{V}$ have been detected by the object detection network [18]. For a particular vehicle $v \in \mathcal{V}$, we let X_i^v denote the coordinates of the i^{th} keypoint of the vehicle in 3D. Also, we parametrize the local ground plane beneath v by its normal vector n_g^v and the distance of the plane from the camera origin d_g^v. Also, we denote by n_c^v the normal of the car. The normal of the car is defined as the normal of a plane that *best*[3] fits the keypoints corresponding to the wheel centers of the cars.

We now formulate a set of cost functions that relax the coplanarity assumptions in [1], [2] and estimate the vehicle's pose and shape as well as the equation of the ground plane patch beneath it.

***Ground Plane Estimation**:* We define a ground plane estimation loss term, which *encourages* the vehicle to be close to the ground plane. Specifically, we obtain the translation vector t_c^v to the bottom of the vehicle v[4] from the camera center. This, in an ideal setting, represents the position vector of a point on the ground plane, the points of which are denoted as X_g^v. Formally, this term (for all vehicles in the image) can be represented as follows,

$$\mathcal{L}_g = \sum_{v \in \mathcal{V}} \|n_c^v \cdot t_c^v - d_g^v\|^2 \tag{4}$$

***Normal Alignment**:* The normal alignment loss term stipulates that the normal of the vehicle (n_c^v) must be encouraged to be parallel to the normal of the estimated ground plane. An initial guess for the ground plane normal is obtained as described earlier, using either lane markings, or a 3-view reconstruction. This loss can be denoted as follows. $\times(.,.)$ denotes the vector cross product.

$$\mathcal{L}_n = \sum_{v \in \mathcal{V}} \| \times (n_c^v, n_g^v)\|^2 \tag{5}$$

***Disambiguation Prior**:* The above loss term has one drawback in that, it is minimized even when the estimated ground plane and vehicle normals are anti-parallel. To disambiguate such unwarranted solutions, we make use of the fact that even the steepest roads in the world have slopes less than $25\,\text{deg}$ [23]. Whenever multiple solutions are available, we encourage the solution that's *more upright* to have a lower cost. If e_2 denotes the Y-axis of the camera coordinate system (i.e., the axis vertically pointing down), we formulate the disambiguation prior as follows (ϵ is a tiny positive constant that provides numerical stability).

$$\mathcal{L}_d = \sum_{v \in \mathcal{V}} \left|\left| \frac{-1}{e_2 \cdot n_c^v + \epsilon} \right|\right|^2 + \left|\left| \frac{-1}{e_2 \cdot n_g^v + \epsilon} \right|\right|^2 \tag{6}$$

***Base Point Priors**:* We also use a loss term that encourages points along the base of the car (this includes keypoints on the car wheel centers, bumpers, etc) to lie as close to the estimated ground plane as possible. If X_b is a keypoint on the car base, and $\mathcal{K}_b$ denotes the set of all keypoints that lie along the base of the car, base point priors are imposed using the following expression.

$$\mathcal{L}_b = \sum_{v \in \mathcal{V}} \sum_{X_b \in \mathcal{K}_b} \|n_c^v \cdot t_c^v - n_c^v \cdot X_b\|^2 \tag{7}$$

***Global Consistency**:* Although we assume that each vehicle has its own planar ground patch, it is safe to assume that road planes are not susceptible to abrupt change. This is encoded into the global consistency loss term, that encourages the planar ground patch of a vehicle to be consistent with that of other vehicles around it. If $\mathcal{V}^n$ denotes the set of all vehicles within a distance d around vehicle v (v is usually chosen to be $5-7$ meters), the global consistency loss term is as follows.

$$\mathcal{L}_c = \sum_{v \in \mathcal{V}} \sum_{v^n \in \mathcal{V}^n} \|n_g^v - n_g^{v^n}\|^2 + \|d_g^v - d_g^{v^n}\|^2 \tag{8}$$

[3]Although, in practice, all 4 wheel centers of a car are coplanar, it may still be numerically hard to determine a plane equation that satisfies all 4 points. So, we fit a plane in the least squares sense to the 4 wheel centers.

[4]We first obtain the rigid-body transform to the origin of the vehicle coordinate frame, and then concatenate to it the rigid-body transformation from the origin of the vehicle coordinate frame to the bottom of the vehicle.

Dimension Regularizers: We also place priors on dimensions of vehicles that we observe, which provides a well-conditioned problem to work with and leads to better convergence rates. We use regularizers similar to ones proposed in [2], and denote the loss term by $\mathcal{L}_{reg}$.

Overall Optimization Problem: The overall minimization problem involving all the energy terms can be posed as follows (cf. Eq 2 4 5 6 8 7).

$$\begin{aligned}\min_{R,t,\lambda,n_g^v,d_g^v,n_c^v} \mathcal{L}_{total} &= \eta_r\mathcal{L}_r + \eta_g\mathcal{L}_g + \eta_n\mathcal{L}_n \\ &+ \eta_d\mathcal{L}_d + \eta_b\mathcal{L}_b + \eta_c\mathcal{L}_c + \eta_{reg}\mathcal{L}_{reg}\end{aligned} \tag{9}$$

Here, η_r, η_g, η_n, η_d, η_b, η_c, and η_{reg} are weighing factors that control the relative importances of each of the loss terms. In practice, η_r, η_g, η_d, and η_b are more dominant compared to the other terms. The actual values of these weighing factors do not really matter as long as the above terms are properly weighted.

The above problem is minimized using Ceres Solver [24], a nonlinear least squares minimization framework, using a Levenberg-Marquardt optimizer with a Jacobi preconditioner. In addition, each term is composed with a Huber loss function, to reduce the effect of outliers on the solution.

IV. Experiments and Results

We perform a thorough quantitative and qualitative analysis of our approach on challenging sequences from KITTI Tracking [7] and SYNTHIA-SF [8] benchmarks. These sequences are chosen such that they capture a diverse class of road plane profiles viz. uphill, downhill, combinations of them, and even banked road planes. We compare the 3D localization error of the proposed method with the current state-of-the-art monocular competitor [2], and demonstrate significant improvements. Through a series of systematic evaluations, we demonstrate that ground plane estimation is vital for accurate localization on roads surfaces with pitch and banks. We also demonstrate that our method is independent of the road plane profile on which vehicles are to be reconstructed. In other words, unlike others (such as [1], [3], [13]) we do not assume that the ego car and the car to be reconstructed are on the same road plane.

Dataset: We use the KITTI Tracking [7] benchmark to evaluate our proposed method. Sequences numbered 1, 3, 7, 8, 9, 10, 11 and 20, which contain a large number of vehicles located on roads with varying plane profiles, were used for evaluating our approach. But, KITTI [7] has only a limited number of steep slopes and banks. So, we also select about 200 vehicles located on challenging plane profiles from sequences numbered 1, 2, 4, 5 and 6 of the SYTHIA-SF [8] dataset. To ensure fair comparison, we evaluate the previous best monocular competitor [2] on the same sequences.

Keypoint Network Training: The proposed network was trained on the Torch framework [25] with more that 1.2 million images generated synthetically using the modified render pipeline presented in [16]. A train-validation split of $75-25$ % was used. The keypoint network was trained for 7 epochs on NVIDIA GTX TITAN X GPUs ($\sim$ 36 hours).

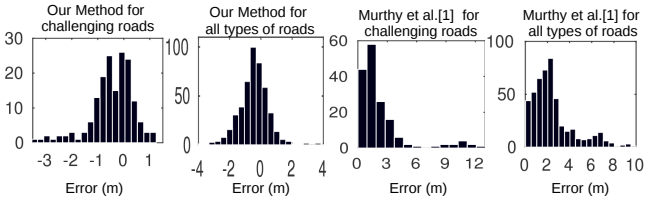

Fig. 6. Histograms showing distribution of localization errors; challenging roads mean slopes, slanted roads, banked roads, etc.

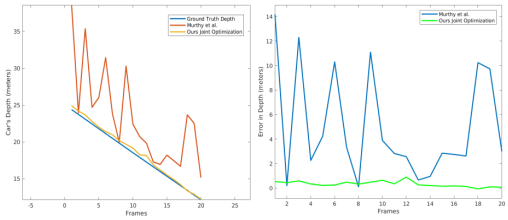

Fig. 7. *Left*: Predicted depth of a car on a steep slope. We compare our method's predictions with [2] against the ground truth. *Right*: Localization error for the same car using the proposed method and [2].

A. Localization Accuracy

To evaluate localization precision, we compute the mean Absolute Translational Error (ATE) of the vehicles (in meters) of the approaches considered against the available ground truth information. We present these results in Table I, Table II and Table III. While Table I captures the overall performance of our approach on KITTI [7] dataset, Table II presents an analysis of the performance of our approach on KITTI[7] sequences with cars on roads with some pitch or banking angle, or parked on pavements. In Table III, we perform a thorough analysis of our approach on SYNTHIA-SF [8] which has extremely steep roads, and demonstrate the efficacy of the proposed approach in adapting to a wide variety of road plane profiles.

We outperform the current best monocular localization result of [2] on the KITTI benchmark [7] by a significant margin. It is important to note that in [2], the shape priors comprised 14 keypoints per vehicle, whereas we use a different shape prior model comprising 36 keypoints per vehicle. However, to emphasize that this improvement does not stem from more expressive shape prior used in this work, we re-implement the approach in [2] using our learnt shape priors and provide an ablation study to further drive the point home. This highlights the importance of the inclusion of ground plane in localization. As shown in Table I, we achieve a mean localization error of 0.86 meters, as compared to 2.61 meters in [2]. This is a mark improvement stemming from the inclusion of ground plane.

We also address challenging sequences with road slopes on KITTI [7] and provide our localization errors in Table II, and perform an ablation study of our approach to highlight how our the inclusion of ground plane reduces the localization error to 0.67 meters, as compared to an error of 2.55 meters given by [2]. The current state-of-the-art [2] relies on the assumption that the plane of the target vehicle and ego vehicle are co-planar. We circumvent this assumption leading to a highly accurate localization of the target vehicle, in a more diverse set of scenarios. For vehicles that are close to

TABLE I

MEAN LOCALIZATION ERROR (STANDARD DEVIATION IN PARENTHESIS) IN METERS FOR THE VEHICLES EVALUATED USING OUR APPROACH ON THE KITTI [7] TRACKING DATASET (HERE ($<x$ m) AND ($>x$ m) DENOTE THE SET OF ALL CARS WITHIN A GROUND-TRUTH DISTANCE OF x METERS AND BEYOND THE DEPTH OF x METERS RESPECTIVELY)

Approach	Overall (m)	$<= 15m$	$<= 30m$	$>30m$
Murthy et. al. [2]	2.61 ($\pm$2.23)	1.59 ($\pm$0.96)	2.52 ($\pm$2.16)	4.30 ($\pm$2.83)
Ours (with coplanarity assumption)	1.00 ($\pm$0.77)	0.67 ($\pm$0.50)	0.94 ($\pm$0.69)	2.19 ($\pm$1.18)
Ours (joint optimization)	**0.86** ($\pm$**0.87**)	**0.55** ($\pm$**0.50**)	**0.79** ($\pm$**0.79**)	**2.16** ($\pm$**1.18**)

TABLE II

MEAN LOCALIZATION ERROR (STANDARD DEVIATION IN PARENTHESIS) IN METERS FOR THE VEHICLES WITH CHALLENGING ROAD PROFILES EVALUATED USING OUR APPROACH ON THE KITTI [7] TRACKING DATASET

Approach	Overall (m)	$<= 15m$	$>15m$
Murthy et. al. [2]	2.55 ($\pm$3.16)	2.32 ($\pm$2.21)	2.92 ($\pm$3.38)
Ours (with coplanarity assumption)	0.95 ($\pm$0.89)	0.92 ($\pm$0.68)	1.00 ($\pm$0.96)
Ours (joint optimization)	**0.67** ($\pm$**0.66**)	**0.64** ($\pm$**0.60**)	**0.72** ($\pm$**0.71**)

TABLE III

MEAN LOCALIZATION ERROR (STANDARD DEVIATION IN PARENTHESIS) IN METERS FOR THE VEHICLES (INCLUDING CHALLENGING ROAD PROFILE) EVALUATED USING OUR APPROACH ON THE SYNTHIA-SF [8] DATASET

Approach	Overall (m)	$<= 15m$	$<= 30m$	$>30m$
Murthy et. al. [2]	76.34 ($\pm$94.03)	54.21 ($\pm$47.93)	66.28 ($\pm$88.74)	86.40 ($\pm$99.32)
Ours (with coplanarity assumption)	32.03 ($\pm$45.60)	6.3 ($\pm$19.17)	21.76 ($\pm$65.76)	42.31 ($\pm$25.42)
Ours (joint optimization)	**0.92** ($\pm$**0.93**)	**0.66** ($\pm$**0.49**)	**0.82** ($\pm$**0.76**)	**1.23** ($\pm$**1.11**)

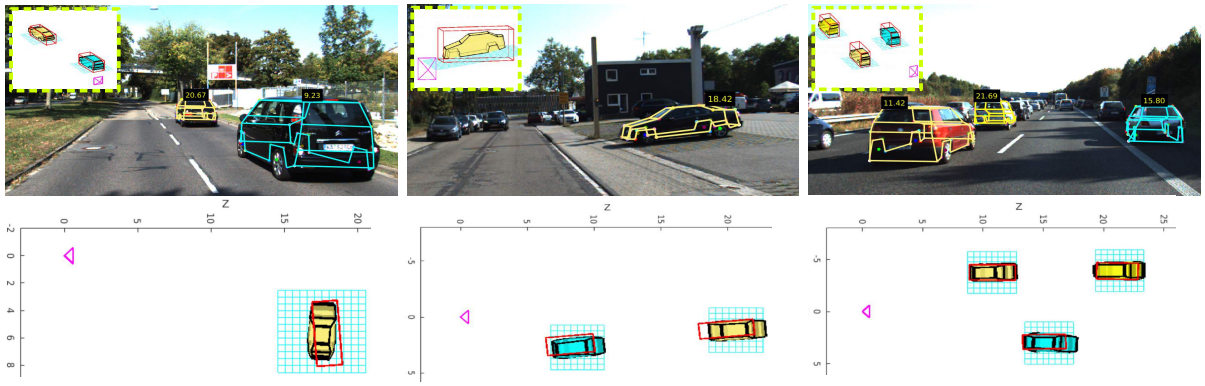

Fig. 8. Qualitative results on KITTI[7]. *Top:* Estimated 3D wireframes (shapes) for selected cars projected on the image, with depth displayed on top of each car and 3D view shown as inset image. *Bottom:* Bird's eye view of the cars overlaid with their respective ground truth bounding boxes (in red).

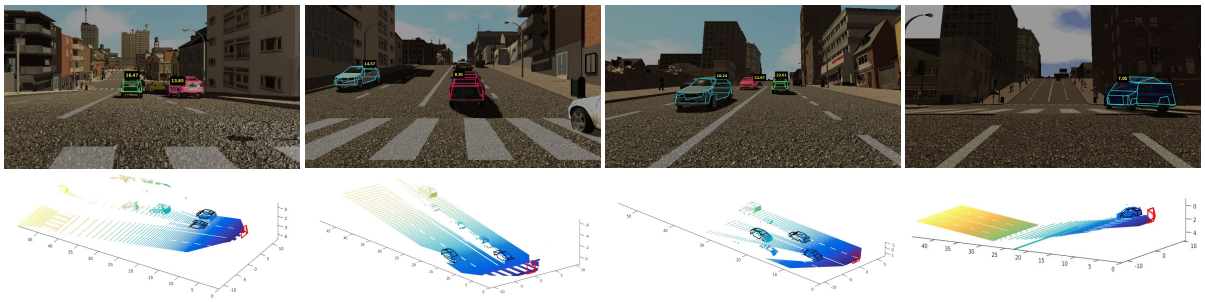

Fig. 9. Qualitative results on SYNTHIA-SF[8]. *Top:* Estimated 3D wireframes for selected cars (on different road profiles) projected on the image, with depth displayed on top of each car. *Bottom:* visualization of the estimated wireframe in 3D, overlaid on dense ground truth 3D scene points.

the car, we achieve a high degree of precision (mean error of about 0.67 meters, with a low standard deviation as well).

To further evaluate our approach, we test it on the extremely challenging SYNTHIA-SF [8] dataset which has steep road surfaces with several non-planar profiles. [1] fails completely in the task of accurate shape estimation and localization of objects in such scenarios, due to the coplanarity assumption. Moreover, the method given by

[2] fails drastically in non-planar surfaces, giving a mean localization error of 76.34 meters, amplified by the sparse set of keypoints (14 keypoints are used, as opposed to ours, which uses 36) leading to large localization errors. Our system achieves a mean localization error of 0.92 meters, the results of which are shown in Table III. The proposed method generalizes well to different plane profiles and performs significantly well. Once again, we stress the importance of ground plane and exhibit how its inclusion helps us to perform significantly better as compared to the approach of [1], which assumes coplanarity of the vehicles and ego car. Fig. 6 shows the error distribution of our approach (first two) and for the approach proposed in [1] (last two); Fig. 7 shows the trajectory and localization error for a car in KITTI [7].

B. Qualitative Results

We showcase the qualitative results of our approach on challenging KITTI [7] and SYNTHIA-SF[8] scenes with moderate to high slopes. For KITTI [7], in Fig. 8, we overlay the final estimate of the car in 3D along with the ground truth 3D bounding box to show how our approach estimates the vehicle shape and pose accurately. For SYNTHIA-SF[8], in Fig. 9, we overlay the estimate of the car after shape and pose adjustment on the ground truth scene points to highlight the accurate shape and pose estimation of the car.

C. Summary of Results

The cornerstone of this effort was to highlight that the presence of non-planar road profiles leads to an unsuccessful pose estimation of cars in urban scenarios by the current state-of-the-art approach, due to the fact that it relies on the coplanarity of the ego vehicle and the car. Our proposed approach is independent of the plane profile on which the car is located. We improve localization accuracy by a large margin through the joint estimation of ground plane in KITTI [7] sequences regardless of whether or not they contain slopes. (cf. Table I and Table II). The importance of the proposed approach is highlighted in Table II, where we achieve a performance boost of about 4 times in scenes with moderate slopes. For an overall comparison on KITTI [7], we evaluate our approach on scenes with different planar and non-planar road surfaces and show an improvement of about 3 times. We further present the performance of our approach on SYNTHIA-SF [8] which has extremely steep roads, resulting in a catastrophic failure of the current state-of-the-art monocular shape and pose estimation [1]. Our performance is significantly improved in such scenes, irrespective of the road profiles, the results of which are reported in Table III. We also perform an ablation study, reported in Table I, Table II and Table III, to highlight the importance of our ground plane estimation policy, and show that it provides a significant performance boost over just the utilization of a well-constrained 36 keypoint shape prior.

V. Conclusions

We presented an approach for accurate 3D localization and shape estimation of vehicles on steep road surfaces. Where, most current monocular localization systems assume coplanarity of the vehicle to be localized and the ego car, we get around this requirement by incorporating and jointly estimating ground plane cues. We validate this claim by showcasing significant improvements over the state-of-the-art monocular localization methods. Heavy traffic situations - where not much of the road surface is visible - are a failure mode for the current approach, and we solicit future work in this direction.

References

[1] J. K. Murthy, G. S. Krishna, F. Chhaya, and K. M. Krishna, "Reconstructing vehicles from a single image: Shape priors for road scene understanding," in *ICRA*, 2017.

[2] J. K. Murthy, S. Sharma, and K. M. Krishna, "Shape priors for real-time monocular object localization in dynamic environments," in *IROS*, 2017.

[3] S. Song and M. Chandraker, "Joint sfm and detection cues for monocular 3d localization in road scenes," in *CVPR*, 2015.

[4] G. P. e. a. Stein, "Vision-based acc with a single camera: Bounds on range and range rate accuracy," in *Intelligent Vehicles Symposium*. IEEE, 2003.

[5] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *ECCV*. Springer, 2016.

[6] M. Z. Zia, M. Stark, and K. Schindler, "Towards scene understanding with detailed 3d object representations," *IJCV*, 2015.

[7] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *CVPR*, 2012.

[8] D. Hernandez-Juarez, L. Schneider, A. Espinosa, D. Vazquez, A. M. Lopez, U. Franke, M. Pollefeys, and J. C. Moure, "Slanted stixels: Representing san francisco's steepest streets," in *BMVC*, 2017.

[9] S. Tulsiani, A. Kar, J. Carreira, and J. Malik, "Learning category-specific deformable 3d models for object reconstruction." *PAMI*, 2016.

[10] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, *et al.*, "Shapenet: An information-rich 3d model repository," *arXiv preprint arXiv:1512.03012*, 2015.

[11] X. Zhou, M. Zhu, S. Leonardos, K. G. Derpanis, and K. Daniilidis, "Sparseness meets deepness: 3d human pose estimation from monocular video," in *CVPR*, 2016, pp. 4966–4975.

[12] S. Song and M. Chandraker, "Robust scale estimation in real-time monocular sfm for autonomous driving," in *CVPR*, 2014, pp. 1566–1573.

[13] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun, "Monocular 3d object detection for autonomous driving," in *CVPR*, 2016, pp. 2147–2156.

[14] F. Chausse, R. Aufrere, and R. Chapuis, "Recovering the 3d shape of a road by on-board monocular vision," in *ICPR*, 2000.

[15] J. Fritsch, T. Kühnl, and F. Kummert, "Monocular road terrain detection by combining visual and spatial information," *IEEE Transactions on Intelligent Transportation Systems*, 2014.

[16] J. K. M. K. M. K. Parv Parkhiya, Rishabh Khawad and B. Bhowmick, "Constructing category-specific models for monocular object slam," in *ICRA*, 2018.

[17] H. Su, C. R. Qi, Y. Li, and L. J. Guibas, "Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views," in *ICCV*, December 2015.

[18] J. X. J. W. J. QiongYan and Y.-W. LiXu, "Accurate single stage detector using recurrent rolling convolution," in *CVPR*, 2017.

[19] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *PAMI*, vol. 39, no. 12, pp. 2481–2495, 2017.

[20] R. K. Satzoda and M. M. Trivedi, "Vision-based lane analysis: Exploration of issues and approaches for embedded realization," in *CVPR Workshops*. IEEE, 2013, pp. 604–609.

[21] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, "DeepFlow: Large displacement optical flow with deep matching," in *ICCV*, Sydney, Australia, Dec. 2013. [Online]. Available: http://hal.inria.fr/hal-00873592

[22] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.

[23] "Kiwi climb: Hoofing up the world's steepest street," http://edition.cnn.com/travel/article/worlds-steepest-street-residents/index.html.

[24] S. Agarwal, K. Mierle, and Others, "Ceres solver," ceres-solver.org.

[25] R. Collobert, K. Kavukcuoglu, and C. Farabet, "Torch7: A matlab-like environment for machine learning," in *BigLearn, NIPS Workshop*, 2011.