Shape Priors for Real-Time Monocular Object Localization in Dynamic Environments

J. Krishna Murthy[1], Sarthak Sharma[1], and K. Madhava Krishna[1]

***Abstract*— Reconstruction of dynamic objects in a scene is a highly challenging problem in the context of SLAM. In this paper, we present a real-time monocular object localization system that estimates the shape and pose of dynamic objects in real-time, using video frames captured from a moving monocular camera. Although the problem seems to be ill-posed, we demonstrate that, by incorporating prior knowledge of the object category, we can obtain more detailed instance-level reconstructions. As opposed to earlier object model specifications, the proposed *shape-prior* model leads to the formulation of a Bundle Adjustment-like optimization problem for simultaneous shape and pose estimation.**

Leveraging recent successes of Convolutional Neural Networks (CNNs) for object keypoint localization, we present a CNN architecture that performs precise keypoint localization. We then demonstrate how these keypoints can be used to recover 3D object properties, while accounting for any 2D localization errors and self-occlusion. We show significant performance improvements compared to state-of-the-art monocular competitors for 2D keypoint detection, as well as 3D localization and reconstruction of dynamic objects.

I. INTRODUCTION

Despite being the holy grail for roboticists for long, SLAM in dynamic environments remains largely unsolved. All state-of-the-art SLAM systems [1], [2], [3], [4] handle dynamic objects by filtering them using standard outlier rejection schemes. With the recent surge in interest for autonomous driving applications, SLAM in presence of moving vehicles has become a desirable component for higher level inference in road scene understanding applications. Autonomous driving platforms are usually equipped with LiDAR, as well as stereo cameras, which are usual sensing options in a SLAM setup. However, it is challenging and interesting to exploit the potential of cheap, off-the-shelf monocular cameras for dynamic, object-based SLAM.

Simultaneous estimation of shape and pose of objects from a moving monocular camera is inherently ill-posed [5], [6]. However, guided by the motivation that humans seem to infer these concepts, owing to their vast prior knowledge, we propose to endow SLAM systems with similar capabilities. We achieve this by making use of *shape priors* to capture the variations in shape of a particular object category. These shape priors are learnt offline, over a small annotated dataset consisting of instances sampled from the category. During inference, we demonstrate the usefulness of these shape priors in the formulation of an optimization problem that can recover the pose and shape of a vehicle in real-time. The formulated optimization problem produces valid results even when the input sequence consists of only a single image [7], and hence naturally falls into an object-SLAM framework.

Fig. 1. Example output from the proposed monocular object localization system. The system is capable of estimating the shape and pose of dynamic objects in real time. The image shows the estimated shapes (wireframes) projected onto the image. Above each of the wireframes is a depth estimate to the object. The inset plot shows the top view of the localization output (red) overlaid on the ground truth (green). Even objects 50 meters are accurately localized.

Leveraging the recent successes of Convolutional Neural Networks (CNNs), a number of systems [8], [9], [10], [11] have been proposed that attempt to infer 3D pose of object categories, using discriminatively trained semantic part locations (*keypoints*) as evidence. Although existing CNN architectures [12], [13], [7], [14] localize keypoints fairly well, they fail to capture pairwise relations among various keypoints, when enough training data is not available. Guided by this, and by the motivation that keypoint visibilities are highly correlated with the viewpoint, we train a single network that predicts keypoints, while capturing consistent pairwise inter-keypoint relationships.

Using the keypoint estimates obtained from the CNN, we formulate a multi-view adjustment problem to recover the 3D locations of the object in each frame. This circumvents problems with state-of-the-art monocular SfM systems for outdoor scenes [5], [15], which rely on sparse matches and usually fail when a high fraction of scene objects are dynamic. The proposed system, on the other hand, can run on arbitrarily long (or short) sequences without collapsing, as we rely on discriminatively trained feature points. The approach has several runtime flavors typical of a visual localization system and can operate in batch mode, incremental mode, or in a sliding window mode.

Contributions:

- We present a novel method of incorporating dynamic objects into a monocular localization framework, by using *shape priors*, that capture the 3D shape of an object category.

[1]J. Krishna Murthy, Sarthak Sharma, and K. Madhava Krishna are with Robotics Research Center, KCIS, International Institute of Information Technology, Hyderabad, India. krrish94@gmail.com, sarthak.sharma@research.iiit.ac.in, mkrishna@iiit.ac.in. This work was supported by grants made available by Qualcomm Innovation Fellowship India, 2017.

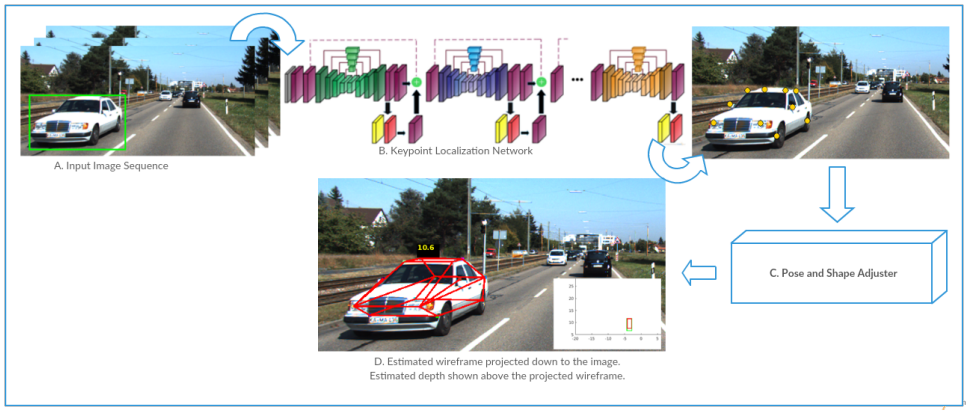

Fig. 2. Illustration of the proposed pipeline. **Clockwise from Top-Left:** The system takes as input an image sequence with 2D object bounding boxes detected. Each of the bounding boxes are then processed by the proposed keypoint localization CNN to obtain 2D locations of a discriminative set of semantic parts. These locations are then incorporated into the proposed multi-view shape and pose adjustment scheme to estimate 3D properties (pose, shape) of the object.

- We propose a solution to circumvent catastrophic failure that SLAM systems experience when a large fraction of the scene is dynamic. We avoid this collapse by training a CNN to precisely localize a discriminative set of features, rather than relying on matches from handcrafted feature descriptors [1].
- We propose a lightweight optimization pipeline that refines the initial estimates to localize dynamic objects in real-time, taking in only a sparse set of feature matches, and robust to self-occlusion.

Evaluation: We perform an extensive analysis of the proposed approach on the KITTI [16] benchmark for autonomous driving. We evaluate our approach on about $2,000$ frames of recorded autonomous driving scenarios and demonstrate superior performance with respect to published monocular competitors. We also perform an extensive evaluation of our proposed CNN architecture for semantic keypoint localization and show an improvement of more than 12% PASCAL-3D dataset. [1]

II. RELATED WORK

3D Properties from a Single Image

Estimating 3D viewpoint or 3D shape from a single image has seen a lot of work [17], [10], [7], [18], [9], [19] in the last couple of years, especially with the availability of large-scale datasets such as ShapeNet [20], PASCAL-3D [21], etc.

Most approaches [7], [9], [10] follow a conventional 2D-to-3D estimation pipeline. First a set of keypoint locations on the 2D (RGB) image is estimated. Then, using a prior shape model [7], [10], or by using a dictionary of poses [9], a deformation/alignment problem is formulated that outputs the 3D structure best explaining the 2D evidence (localized keypoints). In all these cases,explicit 3D keypoint estimates are not required, as they are marginalized out in the estimation process, as highlighted in [9].

Contrary to these, Zia et al [18] propose an end-to-end system that output viewpoint, 2D keypoints, 3D keypoints, as well as keypoint occlusion information. Synthetic models available from ShapeNet [20] are used to train a deep network for the task. Although detection performance under occlusion/truncation is improved by a large margin (over prior art), the synthesized data fails to capture real-world occlusion patterns. Also, the output coordinates are in a canonical frame of reference. On the other hand, the proposed approach optimizes directly in the metric camera coordinate system, to obtain estimates that can be incorporated into a higher level system, such as a trajectory planner or a cruise controller.

While all these approaches estimate the shape/pose of objects from a single image, they do not provide accurate metric localization estimates sutitable for SLAM systems. Moreover, in the context of autonomous driving, we can readily exploit temporal information to obtain better predictions.

Keypoint Localization

Recent successes in single-image shape estimation can be attributed to the availability of deep keypoint localization architectures. One of the earlier approaches for keypoint localization has been presented in [12]. Keypoint estimates from two different scales are composed along with a viewpoint prior to produce keypoint likelihoods across the image. However, the response maps from the CNN were highly multi-modal. As a consequence, accuracy suffered.

In [22], [13], [7], finetuning subnetworks were proposed to refine the estimates from a coarse-grained regressor. In [18], intermediate shape concepts are provided to better supervise the learning process.

Recently, stacked hourglass networks [14] have been proposed for the task of keypoint localization for human pose estimation. These networks are, by construction, multi-scale and possess an iterative refinement nature. We choose this as our base architecture and enforce spatial constraints among keypoints, for better performance.

Reconstructing Moving Vehicles from a Monocular Sequence

Philosophically, the closest work to ours is the one by Falak et al [5], where a stochastic hill-climbing based optimization scheme is proposed over the shape and pose param-

[1]The code and trained models will be made publicly available.

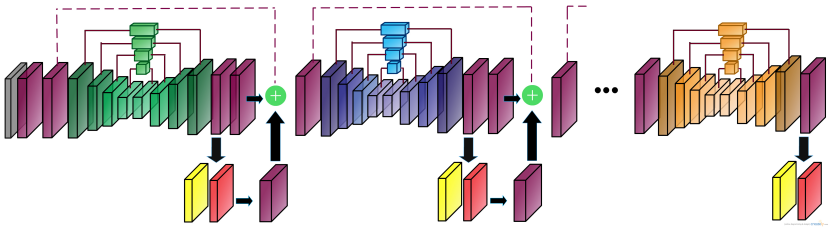

Fig. 3. Proposed network architecture. The yellow and red blocks indicate the keypoint likelihoods and their differences respectively. Joint training over both enables the network to capture pairwise relations among keypoints and serves as additional regularization to prevent overfitting.

eters of a multi-view Active Shape Model (ASM). However, they demonstrate 3D tracking only for short sequences (40-50 frames). The optimization scheme used is also not suitable for real time inference.

Song et al [15], [23] propose a monocular SfM framework for tracking moving vehicles in autonomous driving scenarios. However, they represent cars as 3D bounding boxes and track these bounding boxes across frames. Moreover, they rely on feature matching and optical flow to obtain tracks across long sequences. We use a more detailed shape representation compared to a 3D bounding cuboid. We also use discriminatively trained features to avoid catastrophic failure, and enable long-term tracking.

III. OUR APPROACH

We introduce a novel way of characterizing objects in a SLAM framework. This section presents the proposed object characterization, the CNN architecture for keypoint localization, and finally the backend optimization for the object localization system. Fig. 2 illustrates the overall picture of the proposed pipeline.

To demonstrate our approach, we take up the scenario of autonomous driving, where our target objects for reconstruction are vehicles (predominantly cars and minivans).

A. *Shape Priors*

We encode domain knowledge about the 3D shape of an object category in what we call a *shape prior*. We define the shape of an object to be an ordered collection of semantic *keypoints*. We hypothesize, as in [24], [8], [7], that the shape of a particular instance can be expressed as the sum of the mean shape of the object category and a linear combination of certain *basis shapes*. Intuitively, this means that the keypoints of an object do not deform arbitrarily from one instance to another; rather they span a much lower dimensional subspace of the entire space of possible shapes. Formally, a shape basis is defined as a mean shape $\bar{\mathbf{S}}$ and a set of B basis shapes $\mathbf{V}_k$ ($k = \{1..K\}$), such that the shape of any new instance $\mathbf{S}$ can be expressed as

$$\mathbf{S}_m = \bar{\mathbf{S}} + \sum_{j=1}^{B} \lambda_j \mathbf{V}_j \tag{1}$$

In 1, λ_j is the weight of the jth basis shape. These basis shapes can be learnt entirely from 2D images, as demonstrated in [7], [8], or from 3D CAD models, as presented in [25]. We follow the method of [7] and learn the shape priors over a 2D keypoint annotated dataset consisting of about 300 images from the PASCAL3D [21] dataset.

B. *Keypoint Localization CNN*

Using traditional feature extraction methods, it is very hard to obtain consistent feature matches on dynamic objects across long sequences [15], [5]. Hence, we train a stacked hourglass CNN architecture to accurately localize the chosen semantic *keypoints*.

Fig. 3 illustrates the network architecture. Unlike existing architectures for keypoint localization [12], [13], [22], the stacked hourglass maintains fixed spatial dimensions (height, width) across the network. It takes in a $3 \times 64 \times 64$ image of the resized, cropped bounding box containing a car, as input. The core component of the network is what we call an *hourglass* [14], which consists of a symmetric encoder and a decoder block. To compensate for the loss of information due to pooling in the encoder block, a set of skip connections forward data (via a series of convolutions) to the corresponding decoder block. After each such hourglass, the network outputs a set of keypoint likelihood maps (one map per keypoint) over the entire image. Multiple such hourglass modules are stacked on top of each other to iteratively refine the keypoint likelihoods. Predictions from one hourglass are fed into the network via a 1×1 convolution block. An *intermediate* loss function is applied to the network output at the end of each hourglass. This kind of intermediate supervision has shown to perform better than scenarios where loss has been applied only at the end of the network [14], [18].

CRF-Style Stacked Hourglass Networks

To explicitly force the network to learn pairwise keypoint distance relations, we propose a *CRF-Style* loss function which is applied to the predictions at the end of each hourglass. Given K keypoint pairs, ${}^{K}C_2$ combinations seem likely. However, we note that pairwise keypoint distance is transitive. For instance, if the pairwise distance between keypoints i and j, as well as keypoints i and k, is enforced, the pairwise distance between keypoints j and k enforces itself implicitly. So, it is enough if we have K pairwise potentials, which keeps the dimensionality of the pairwise terms linear in the number of keypoints.

Specifically, apart from enforcing that constraint that each keypoint likelihood must be precisely localized, we enforce the constraint that inter-keypoint distances must be correct. To accomplish this, for each of the K keypoints, we compute the *difference maps*, i.e., for a given hourglass $\mathcal{H}$, if h_i denotes the ith heatmap, we denote the difference heatmap as $\Delta_i = h_i - h_1$ $\{i = 1..K\}$. Formally, we have a unary

potential Φ and a binary potential Ψ such that, for each example x_i in the training set $i \in \{1..N\}$, we minimize the sum of the following functions simultaneously.

$$\begin{aligned}\Phi(\mathbf{x}) &= \sum_{i=1}^{N}\sum_{k=1}^{K} \|h_k(x_i) - h_k^{GT}(x_i)\|^2 \\ \Psi(\mathbf{x}) &= \sum_{i=1}^{N}\sum_{k=1}^{K} \|\Delta_k(x_i) - \Delta_k^{GT}(x_i)\|^2\end{aligned} \tag{2}$$

In Eq 2, $h_k^{GT}(x_i)$ and $\Delta_k^{GT}(x_i)$ represent the ground truth keypoint likelihood and difference of keypoint likelihoods respectively for the kth keypoint of the ith training sample. If a keypoint is occluded, then its corresponding ground-truth likelihood is zero across the entire image.

The network is trained end-to-end, minimizing the sum of the unary and binary potentials via mini-batch stochastic gradient descent.

C. *Object Localization Formulation*

In our object localization formulation, we use the learnt Shape Priors to formulate a Bundle Adjustment-like optimization problem to simultaneously estimate the shape and pose of an object, given 2D keypoint localizations across a sequence of image frames.

Problem Specification

We assume that a vehicle (stationary/moving) has been detected (in 2D) over a sequence of F frames. In each frame, each of the K keypoints have been localized by the keypoint network. Throughout, we assume that i is an index over keypoints ($i \in \{1..K\}$) and that f is an index over views (frames) ($f \in \{1..F\}$). Given a set of 2D observations of keypoint locations $\mathbf{x}_i^f$, recover the 3D shape and pose of the object, i.e, estimate $\mathbf{X}_i^f$.

Note that directly estimating $\mathbf{X}_i^f$ is an ill-posed problem [7], as this will allow for arbitrary deformations in the object shape. We instead estimate the shape parameters λ_j ($j \in \{1..B\}$), where B is the number of shape deformation bases in Eq 1. We also estimate the pose parameters $\mathbf{R}^f$ (rotation), $\mathbf{t}^f$ (translation), such that

$$\mathbf{X}^f = \mathbf{R}^f \left(\bar{\mathbf{X}}^f + \sum_{j=1}^{B} \lambda_j * V_j \right) + \mathbf{t}^f \tag{3}$$

In Eq 3, $\bar{\mathbf{X}}^f$ refers to the mean shape of the vehicle.

Pose and Shape Adjustment

To estimate the shape and pose of a vehicle over a sequence, we optimize for a solution in the maximum likelihood sense, i.e., *that pose and shape are more likely which explain the image evidence (2D keypoints) the best*. For the same, we make use of the pinhole camera model to define a reprojection error that only allows deformations that are in accordance with the class-specific shape prior.

Shape-Constrained Reprojection Error: Concretely, given the camera intrinsics $\mathbf{K}$, we specify the reprojection error term as follows.

$$\mathcal{R} = \sum_{f=1}^{F}\sum_{i=1}^{K} \left\| \mathbf{x}_i^f - \pi \left[\mathbf{K}\mathbf{R}^f \left(\bar{\mathbf{X}}^f + \sum_{j=1}^{B} \lambda_j * V_j \right) + \mathbf{K}\mathbf{t}^f \right] \right\|^2 \tag{4}$$

Temporal Trajectory Consistency: This term imposes a regularizer on the rotation and translation between successive frames from a sequence. If $\omega^{\mathbf{f}}$ is the axis angle vector corresponding to $\mathbf{R}^f$,

$$\mathcal{M} = \sum_{f=2}^{F} \left(\left\| \omega^{f-1} - \omega^f \right\|^2 + \left\| \mathbf{t}^{f-1} - \mathbf{t}^f \right\|^2 \right) \tag{5}$$

Dimension Priors: This term imposes a regularizer on the dimensions of the estimated shape. If $\mathcal{H}(\mathbf{X}^f)$, $\mathcal{W}(\mathbf{X}^f)$, and $\mathcal{L}(\mathbf{X}^f)$ denote the height, width, and length of the wireframe respectively, and if $\bar{\mathcal{H}}$, $\bar{\mathcal{W}}$, and $\bar{\mathcal{L}}$ denote the priors for these dimensions (computed over a training subset),

$$\mathcal{S} = \sum_{f=1}^{F} \sum_{\mathcal{D}\in\{\mathcal{H},\mathcal{W},\mathcal{L}\}} \left\| \mathcal{D}(\mathbf{X}^f) - \bar{\mathcal{D}} \right\|^2 \tag{6}$$

Ground-plane Prior: All objects that we observe are constrained to lie on the Ground Plane. Hence, we can additionally constrain the object rotation to be directed only about the ground plane normal $\mathbf{n}_g^f$. This is done by ensuring that the axis angle vector ω^f (corresponding to $\mathbf{R}^f$) is parallel to the ground plane vector.

$$\mathcal{G} = \sum_{f=1}^{F} \left\| \mathbf{n}_g^f \times \omega^f \right\|^2 \tag{7}$$

Finally, the adjustment problem is specified as follows.

$$\min_{\lambda_j, \omega^f, \mathbf{t}^f} \quad \rho(\mathcal{R}) + \mathcal{M} + \rho(\mathcal{S}) + \mathcal{G} \tag{8}$$

Here, ρ represents an M-Estimator, and is used to reduce the effects of outliers on the estimation procedure. In all our experiments, we use the Tukey biweight M-estimator.

Initialization: We assume that we have the height of the camera above the ground plane (XZ-plane) and use this to initialize a rough estimate of the vehicle position, as in [15], [23], [25]. We also obtain a rough viewpoint estimate from a VGG-like CNN [12].

Self-occlusion and Imprecise Keypoints: We weigh each observation (2D keypoint) by the corresponding confidence score output by the keypoint localization network. The network randomly fills in missing/occluded keypoints, and these are usually discarded as outliers by the M-estimators used for optimization.

Modes of Operation: The proposed pipeline provides all runtime flavors expected of a typical SLAM system, viz. batch mode, incremental mode, and windowed execution. In the limiting case, the system can also be used to recover 3D properties from just a *single image*. However, in that case, pose and shape cannot be jointly estimated [7]. The approach outlined in [7] must be adopted.

Approach	< 20 m	< 25 m	< 30 m	< 45 m	> 45 m
Single-View [7]	**0.45**	0.99	1.37	2.24	5.41
Multi-View (Incremental)	0.46	0.73	1.35	2.01	**4.45**
Multi-View (Batch)	0.46	**0.67**	**1.01**	**1.47**	4.47

TABLE I

LOCALIZATION ERROR (IN METERS) OF ALL VEHICLES EVALUATED USING DIFFERENT MODES OF THE APPROACH.

Approach	< 0.5 m (%)	< 1 m (%)	< 1.5 m (%)	< 2 m (%)
Zia et al [25]	N/A	55.2	76.24	89.38
Falak et al [5]	N/A	70.44	95.08	98.36
Ours (Multi-view, batch mode)	**68.19**	**81.82**	**98.00**	**100.00**

TABLE II

LOCALIZATION ACCURACY (PERCENTAGE OF VEHICLES LOCALIZED BELOW THE THRESHOLD DISTANCE) OF ALL VEHICLES EVALUATED IN [5].

Approach	**Height Error** (%)	**Width Error** (%)	**Length Error** (%)	**Size Error** (Near) (%)	**Size Error** (Far) (%)
Song et al [15]	N/A	N/A	N/A	14.8	12.3
Song et al [23]	N/A	N/A	N/A	7.3	11.8
Ours (incremental)	**6.36**	**6.85**	**8.05**	**6.57**	**7.51**

TABLE III

ERROR IN RECOVERY OF 3D PROPERTIES. THE *Near* AND *Far* CATEGORIES ARE IN ACCORDANCE WITH THE EVALUATION OF [15], [23]. OBJECTS THAT ARE CLOSER THAN $15m$ ARE CONSIDERED *Near*.

IV. RESULTS

We perform a thorough qualitative and quantitative analysis of the proposed approach on several sequences of the challenging KITTI tracking benchmark [16]. The sequences are chosen such that there is sufficient variance in illumination, viewpoint, high fraction of moving vehicles, and a fair mix of near and far vehicles. We compare the 3D localization error obtained by the proposed approach with state-of-the-art monocular competitors [5], [25], [15], [23]. Moreover, to demonstrate the effectiveness of the proposed keypoint localization network, we evaluate the 2D keypoint localization accuracy on the PASCAL3D dataset [21]. Finally, we show qualitative results (Fig. 4) which indicate that the proposed approach works over a wide range of vehicle shapes and poses.

Datasets: We use the KITTI [16] tracking benchmark to evaluate our localization accuracy. Sequences 2, 3, 4, 5, 6, 10, and 12, which contain a large number of moving vehicles, were used for evaluating the approach. The remaining sequences have been used to estimate dataset statistics, used as priors in the optimization pipeline.

Keypoint Network Details: To train the keypoint network, we use keypoint-annotated data for the *car* class of the PASCAL3D [21] dataset. Random horizontal flips, crops, and color space augmentation were employed to synthesize newer samples. The network was trained using the popular Torch framework.

Misc. Implementation Details: The multi-view and single-view adjustment pipelines have been implemented using Ceres Solver [26]. The optimization problem was solved using a dense Schur linear system solver with a Jacobi preconditioner.

A. Localization Accuracy

To analyze the efficiency of object localization, we evaluate the average translation error of the car (in meters) from the ground truth location. This evaluation is presented in Table I.

We test 3 *flavors* of the proposed system. *Single-View* refers to the case where each image is independently processed, and no temporal coherence is exploited. In the *Multi-View (Incremental)* version, we add temporal consistency constraints between a newly added frame and the most recent optimized estimate. In the *Multi-View (Batch)* mode of operation, we assume the entire sequence is available prior to optimization.

As one would expect, the multi-view approach outperforms the other modes of execution, as it has access to more information that can be used to over-constrain the objective function. However, the single-view error is also quite low, except for far-off objects. The multi-view approach is not quite suitable for real-time execution since it assumes all data is available before performing the optimization. Interestingly, the incremental version, which is real-time, performs marginally better than the multi-view (batch) mode for far-off objects.

One recent work that attempts monocular reconstruction of moving vehicles is by Falak et al [5]. However, localization accuracy in [5] is evaluated only for vehicles with depths ranging from 4 m to 25 m. Moreover, they require two perpendicular planar surfaces of the car to be visible in order for their moving plane homography framework to produce inter-frame motion estimates. A comparison is provided in Table II.

Clearly, the proposed approach provides precise localization estimates in metric scale and outperforms prior art by a significant margin. Interestingly, none of the cars have a localization error of more than 2 m. Moreover, the proposed approach runs real-time [2], whereas the other approaches incur processing times of about 15 minutes per frame [25], [5].

Table III shows the advantage of the proposed system as compared to approaches that treat cars as 3D bounding boxes. Specifically, the proposed system recovers 3D object

[2]Assuming that a GPU is available to run the keypoint network

properties (height, width, length) more accurately, taking advantage of the shape priors. Since the approaches [15], [23] are real-time, we compare it with the incremental version of the proposed system. Although our batch mode recovers more accurate 3D properties, such a comparison would be unfair.

To further illustrate the capability of our system to localize across a long sequence using only sparse matches, we plot the depth estimates of an object over 241 frames in Fig 5. Initially, when the object is very far-off ($80m$), the system incurs significant estimation errors. However, it soon stabilizes and tracks the object accurately until the end.

Fig 8 shows a few samples of the output obtained from the proposed localization system.

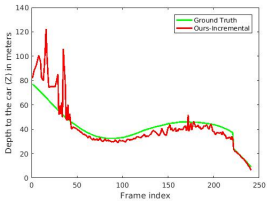

Fig. 5. Localization accuracy over a long sequence (241 frames).

B. Keypoint localization (2D)

Herein, we evaluate the accuracy of our 2D keypoint localization network. To evaluate our network, we use the standard PCK (Percentage of Correct Keypoints) and APK (Average Precision of Keypionts) metrics, used in [27], [12], [14]. In our analysis, we use a very tight threshold of 2 px to determine whether or not our keypoint estimate is correct. We compare the accuracy obtained for the *car* class with the approaches [18], [12].

Approach	PCK (%) ($\alpha = 0.1$)
Tulsiani et al [12]	81.3
Zia et al [18]	81.8
Ours (hourglass, CRF-style loss)	**93.4**

TABLE IV

EVALUATION OF THE PROPOSED KEYPOINT LOCALIZATION NETWORK ARCHITECTURE.

Table IV shows the keypoint localization accuracy obtained by the proposed network architecture. The results indicate a significant performance boost in the task of keypoint localization, which also helps in boosting the performance of the 3D object localization pipeline.

Generalization Performance

To evaluate the generalization capability of various keypoint localization architecture, we evaluate the PCK measure

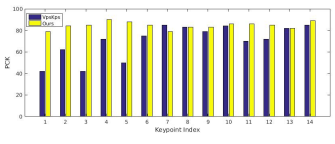

Fig. 6. Per-keypoint PCK comparison for VpsKps [12] and the proposed keypoint network architecture. Parts 1-4 correspond to the wheels, 5-6 correspond to the headlights, 7-8 correspond to the taillights, 9-10 correspond to the side-view mirrors, and 11-14 correspond to four corners of the rooftop.

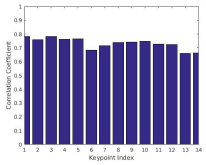

Fig. 7. Correlation coefficient between the keypoint confidence score output by the proposed CNN and the ground-truth visibility $(0-1)$ vector. On an average, the correlation coefficient is 0.72, which indicates that the network has learnt visibility information.

on a keypoint-annotated dataset comprising of about 19000 cars from a subset of the KITTI [16] object dataset, made available by [7]. Fig 6 compares the per-keypoint PCKs of the keypoint network described in [12] compared to the proposed architecture. Both the networks were trained entirely on the same train split of PASCAL3D [21]. However, the proposed architecture performs significantly better than [12] for most of the keypoints.

Correlation with Visibility

The proposed CNN architecture, in addition to localizing keypoints, provides a confidence score for each estimate which determines the likelihood of that keypoint being visible. To analyze this empirically, we compute the Pearson Correlation Coefficients for each keypoint confidence to its ground truth visibility (binary) vector. This is shown in Fig 7. The correlation is quite high (0.72 on an average), which indicates that the CNN has learnt the notion of visibility.

Qualitative Results

Finally, a few qualitative results of keypoint localization are shown in Fig 4.

V. CONCLUSIONS

In this work, we presented an approach for real-time monocular object localization. Although the problem is ill-posed, we demonstrated that prior knowledge about object shapes helps in accurate localization. We proposed a novel method of incorporating this prior knowledge into an object localization system by means of *shape priors*. Further, we proposed a keypoint localization architecture that improves the state-of-the-art for car keypoint localization by more than 12%. The proposed shape characterization naturally falls into a Bundle Adjustment-like optimization framework which can be efficiently solved using only a sparse set of discriminative feature matches. Qualitative and quantitative analysis was performed on multiple sequences from the challenging KITTI [16] tracking benchmark.

REFERENCES

[1] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[2] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *European Conference on Computer Vision*. Springer, 2014, pp. 834–849.

[3] J. Civera, D. Gálvez-López, L. Riazuelo, J. D. Tardós, and J. Montiel, "Towards semantic slam using a monocular camera," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*. IEEE, 2011, pp. 1277–1284.

Fig. 4. Qualitative results showing the 2D keypoint localization performance of the proposed architecture. Top 7 keypoints per instance are shown (in accordance with the confidence scores output by the CNN). Discriminative features are extracted consistently across instances, pose variations, and occlusions. The last row shows some failure cases.

Fig. 8. Qualitative results showing the performance of the proposed system over various sequences of KITTI [16] Tracking. Each image shows a set of estimated wireframes (shapes) projected down to 2D. The inset plot shows the estimated 3D location of a car (red) overlaid on the ground truch (green), for some of the cars in the image.

[4] D. Gálvez-López, M. Salas, J. D. Tardós, and J. Montiel, "Real-time monocular object slam," *Robotics and Autonomous Systems*, vol. 75, pp. 435–449, 2016.

[5] F. Chhaya, D. Reddy, S. Upadhyay, V. Chari, M. Z. Zia, and K. M. Krishna, "Monocular reconstruction of vehicles: Combining slam with shape priors," in *Proceedings of the IEEE Conference on Robotics and Automation*, 2016.

[6] Y. Gao and A. L. Yuille, "Exploiting symmetry and/or manhattan properties for 3d object structure estimation from single and multiple images," *arXiv preprint arXiv:1607.07129*, 2016.

[7] 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 *Proceedings of the IEEE Conference on Robotics and Automation (In Press)*, 2017.

[8] S. Tulsiani, A. Kar, J. Carreira, and J. Malik, "Learning category-specific deformable 3d models for object reconstruction." *IEEE transactions on pattern analysis and machine intelligence*, 2016.

[9] X. Zhou, M. Zhu, S. Leonardos, K. G. Derpanis, and K. Daniilidis, "Sparseness meets deepness: 3d human pose estimation from monocular video," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4966–4975.

[10] J. Wu, T. Xue, J. J. Lim, Y. Tian, J. B. Tenenbaum, A. Torralba, and W. T. Freeman, "Single image 3d interpreter network," in *European Conference on Computer Vision*. Springer, 2016, pp. 365–382.

[11] G. Pavlakos, X. Zhou, A. Chan, K. Derpanis, and K. Daniilidis, "6-dof object pose from semantic keypoints," in *Proceedings of the IEEE Conference on Robotics and Automation (In Press)*, 2017.

[12] S. Tulsiani and J. Malik, "Viewpoints and keypoints," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2015, pp. 1510–1519.

[13] T. Pfister, J. Charles, and A. Zisserman, "Flowing convnets for human pose estimation in videos," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1913–1921.

[14] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *European Conference on Computer Vision*. Springer, 2016, pp. 483–499.

[15] S. Song and M. Chandraker, "Robust scale estimation in real-time monocular sfm for autonomous driving," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1566–1573.

[16] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[17] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun, "Monocular 3d object detection for autonomous driving," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2147–2156.

[18] Q.-H. T. X. Y. G. D. H. M. C. Chi Li, M. Zeeshan Zia, "Deep supervision with shape concepts for occlusion-aware 3d object parsing," *arXiv preprint arXiv:1612:02699*, 2016.

[19] M. Zhu, X. Zhou, and K. Daniilidis, "Single image pop-up from discriminatively learned parts," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 927–935.

[20] 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.

[21] Y. Xiang, R. Mottaghi, and S. Savarese, "Beyond pascal: A benchmark for 3d object detection in the wild," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2014.

[22] Y. Sun, X. Wang, and X. Tang, "Deep convolutional network cascade for facial point detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3476–3483.

[23] S. Song and M. Chandraker, "Joint sfm and detection cues for monocular 3d localization in road scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3734–3742.

[24] L. Torresani, A. Hertzmann, and C. Bregler, "Nonrigid structure-from-motion: Estimating shape and motion with hierarchical priors," *IEEE transactions on pattern analysis and machine intelligence*, vol. 30, no. 5, pp. 878–892, 2008.

[25] M. Z. Zia, M. Stark, and K. Schindler, "Towards scene understanding with detailed 3d object representations," *International Journal of Computer Vision*, vol. 112, no. 2, pp. 188–203, 2015.

[26] S. Agarwal, K. Mierle, and Others, "Ceres solver," http://ceres-solver.org.

[27] Y. Yang and D. Ramanan, "Articulated human detection with flexible mixtures of parts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2878–2890, 2013.