A Neural Network Architecture for Accurate 4D Vehicle Pose Estimation from Monocular Images with Uncertainty Assessment*

Tomasz Nowak(✉) and Piotr Skrzypczyński

Poznan University of Technology, Institute of Robotics and Machine Intelligence, ul. Piotrowo 3A, 60-965 Poznań, Poland
tomasz.nowak@doctorate.put.poznan.pl, piotr.skrzypczynski@put.poznan.pl

Abstract. This paper proposes a new neural network architecture for estimating the four degrees of freedom poses of vehicles from monocular images in an uncontrolled environment. The neural network learns how to reconstruct 3D characteristic points of vehicles from image crops and coordinates of 2D keypoints estimated from these images. The 3D and 2D points are used to compute the vehicle pose solving the Perspective-n-Point problem, while the uncertainty is propagated by applying the Unscented Transform. Our network is trained and tested on the ApolloCar3D dataset, and we introduce a novel method to automatically obtain approximate labels for 3D points in this dataset. Our system outperforms state-of-the-art pose estimation methods on the ApolloCar3D dataset, and unlike competitors, it implements a full pipeline of uncertainty propagation.

Keywords: Vehicle pose estimation · 3D scene understanding · Deep learning

1 Introduction

It is essential for autonomous cars to be able to predict the poses of different objects in the environment, as it allows these agents to localise relatively to other vehicles or infrastructure on the road. Although the pose of another vehicle can be accurately measured using 3D laser scanner data [12], it is practical to obtain good accurate estimates of the pose of a vehicle from a single-camera image [18].

Song *et al.* [25] demonstrated that it is difficult to estimate vehicle poses from monocular images in a traffic scenario, because the problem is ill-posed due to the lack of obvious constraints and because of unavoidable occlusions. In our recent research [21], we showed how a deep neural network derived from the HRNet [29], originally created for human pose estimation, can be used to

* Research Supported by Poznań University of Technology Internal Grant 0214/SBAD/0242

B. Luo et al. (Eds.): ICONIP 2023, CCIS 1962, pp. 396–412, 2024.
https://doi.org/10.1007/978-981-99-8132-8_30

accurately position a single camera in relation to a known road infrastructure object. In this paper, we introduce a novel deep learning architecture, also based on HRNet, which estimates 4D poses (3D Cartesian position and yaw angle) of vehicles in a realistic traffic scenario. This architecture estimates 2D keypoints on RGB image crops and then leverages these point features to estimate 3D coordinates of the characteristic points of a vehicle CAD model. The vehicle pose is computed by solving the Perspective-n-Point problem, given a set of n estimated 3D points of the vehicle model and their corresponding 2D keypoints in the image.

Unfortunately, the dataset we use, ApolloCar3D [25], does not contain associations between the 3D coordinates of the characteristic points of different car models and the annotated 2D keypoints in the images. To overcome this problem we propose also a new approach to automatically pre-process the dataset in order to obtain these associations leveraging the known poses and 3D CAD models of the vehicles seen in the images.

Knowing an estimate of the uncertainty in object pose estimates is crucial for the safe and reliable operation of autonomous cars. Therefore, we propose a two-step uncertainty estimation process. In the first step, we augment the architecture of the proposed neural network to estimate the uncertainty of the obtained keypoints. This uncertainty is represented by covariance matrices and then is propagated in the second step through the PnP algorithm using the unscented transform technique [22], which takes into account the non-linearity of the pose estimation algorithm being applied. Pose estimates with compatible covariance matrices are a unique feature among pose estimation algorithms that employ deep learning.

We thoroughly evaluate our approach on the ApolloD3 benchmark, proposed in the same paper as the dataset we use for training, demonstrating that the results outperform the state-of-the-art methods in terms of accuracy, while they are accompanied by human-interpretable estimates of spatial uncertainty.

The remainder of this paper is organised as follows: The most relevant related works on pose, keypoints, and uncertainty estimation are briefly reviewed in Sect. 2. The structure of the proposed pose estimation system is detailed in Sect. 3, while the approach to supervised training of our neural network, including pre-processing of the dataset, is explained in Sect. 4. Next, techniques applied to estimate and propagate the uncertainty are presented in Sect. 5. Finally, experiments on the ApoloCar3D dataset and the obtained results are presented in Sect. 6, followed by the conclusions given in Sect. 7.

2 Related Work

2.1 Vehicle Pose Estimation

Vehicle pose estimation solutions are of interest to researchers mainly due to the development of autonomous cars. Many solutions to the problem of determining the 3D pose of a vehicle based on observations from depth sensors such as LiDAR have been developed [30,32]. The pose of a vehicle can also be accurately estimated from scene depth data obtained using stereo vision [14]. Estimating the

pose of a vehicle from a single monocular camera image is most difficult, due to the lack of depth data and the difficulty in extracting pose constraints imposed by the environment from the image [25]. Nowadays, the state-of-the-art in estimating vehicle pose from monocular images involves the use of deep learning techniques combined with geometric priors [5]. Two basic types of approaches can be distinguished: direct, without detection of feature points, and indirect, using the detection of vehicle feature points in images.

Older direct pose estimation systems often relied on external neural network models that generated partial solutions, such as 2D proposals in [20], which were then cropped and processed in another network to estimate 3D bounding boxes and their orientations. Also, Xu and Chen [31] used separated neural networks to predict a depth image of the scene converted then to a point cloud and to generate 2D bounding boxes used to sample from this point cloud. More recent approaches, like the M3D-RPN [1] utilise a standalone 3D region proposal network leveraging the geometric relationship of 2D and 3D perspectives and allowing 3D bounding boxes to use features generated in the image-space. Yet differently, [9] exploits class-specific shape priors by learning a low dimensional shape-space from CAD models. This work uses a differentiable render-and-compare loss function which allows learning of the 3D shapes and poses with 2D supervision.

The indirect approaches typically align 3D car pose using 2D-3D matching with 2D keypoints detected on images and the provided CAD models of vehicles [2], exploiting also geometric constraints, e.g. the co-planar constraints between neighbouring cars in [2]. The RTM3D [15] predicts the nine keypoints of a 3D bounding box in the image space, then applies the geometric relationship of 2D and 3D perspectives to recover the parameters of the bounding box, including its orientation. In indirect approaches to vehicle pose estimation an important problem is to localise the occluded keypoints. This problem is addressed by the Occlusion-Net [23], which predicts 2D and 3D locations of occluded keypoints for objects. One of the recent developments, the BAAM system [10], leverages various 2D primitives to reconstruct 3D object shapes considering the relevance between detected objects and vehicle shape priors.

The approach in [17] is similar to ours in using estimated 2D and 3D points, but it directly regresses 20 semantic keypoints that define the 3D pose of a vehicle, whereas we extract more keypoints and compute poses indirectly, solving the PnP problem.

In this work, fewer keypoints are detected and used by the PnP algorithm, whereas the evaluation is done only on the KITTI dataset. The same problem is considered in [24] and, like us, the authors aim to select the best points for pose estimation, but the results they show use ground truth data – either about depth or 2D points.

Similarly to our solution, reprojection loss is used to train the model in the indirect GSNet [7] system, which is also evaluated on the ApolloCar3D dataset. However, the authors focus more on vehicle shape estimation, which is out of the scope of our work.

2.2 Keypoints and Uncertainty

In terms of accurate and robust detection of keypoints in images, significant progress has recently been made in human pose estimation applications [26]. The High Resolution Network (HRNet) [29] is a leading solution for keypoint detection in human pose estimation. As this architecture ensures high resolution of the feature maps through the entire network processing pipeline, which results in accurate location of the keypoints, we have selected the HRNet as a backbone of the neural part of our vehicle pose estimation system. In our previous work [21], we showed that it is possible to adapt HRNet to the task of estimating the feature points of an object with a known CAD model from RGB images. In the same work, we also introduced a method for estimating the spatial uncertainty of feature points in the form of a covariance matrix, inspired by the algorithm for estimating the uncertainty of keypoints in a face recognition task [8]. While there are classical methods for the propagation of uncertainty in computer vision [4], the problem of estimating feature points uncertainty or object pose uncertainty in deep learning based systems has been addressed in relatively few research papers. Several versions of a camera pose estimation method that consider the uncertainty of point and line features while computing the pose are proposed in [27], but these methods do not produce a covariance matrix for the final pose estimate. The geometric (spatial) uncertainty of features in the PnP problem is also considered in [3]. Recently, approaches to pose estimation that consider spatial uncertainty were presented for deep learning-based human pose recognition [13] and general object pose determination from images [16,33]. As far as we know, there are no publications tackling the estimation of the keypoints uncertainty and propagation of this uncertainty to the final vehicle pose for monocular pose estimation in the context of automotive applications.

3 Structure of the Pose Estimation System

The diagram of the whole processing pipeline is shown in Fig. 1. The details of heads architecture are presented in Fig. 2. It accepts as input only an image crop containing the considered vehicle. The processing pipeline consists of several modules: The 2D Keypoint Estimation Head, the 3D Keypoint Estimation Head, the Keypoint Score Head (KSH) for evaluating the accuracy of individual point estimates, and the Uncertainty Estimation Head for estimating the uncertainty of 2D and 3D points.

3.1 Estimation of 2D Keypoints

The first module is a deep neural network that estimates the 2D coordinates of the car's characteristic points on the image. Our approach combines the HRNet48 [29] as a backbone with a head for the feature extraction and heatmap estimation of 66 distinctive points on the image. We utilise input image crops of 256×192 pixels, providing sufficient detail for precise and fast feature extraction. The output consists of 48 feature maps of size $w \times h$, in our models w=64 and h=48. During training, we compute the loss function only for visible points,

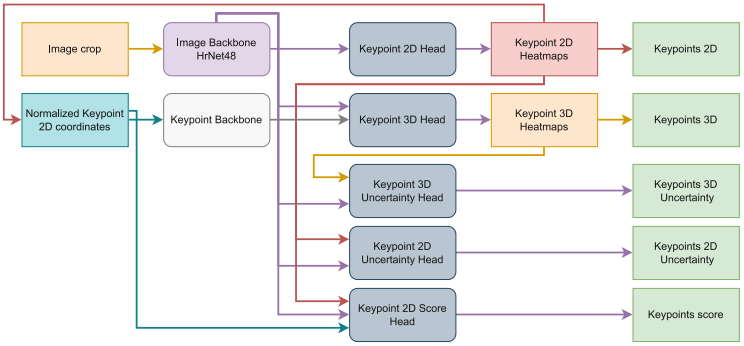

Fig. 1. Architecture of the proposed vehicle pose estimation system

applying a visibility mask that defines which points are evident for each instance. This process ensures that the model focuses on relevant data and is not influenced by obscured or irrelevant points. We employ unbiased encoding and decoding methods described in [6] for the preparation of ground-truth heatmaps for training and the subsequent decoding coordinates from the estimated heatmaps. This technique enhances prediction accuracy by reducing the quantization error.

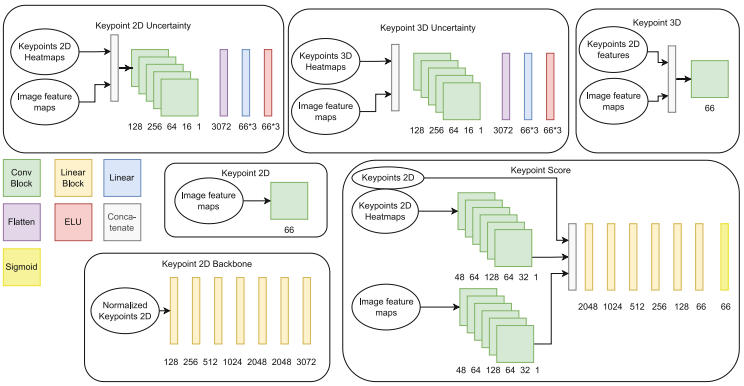

Fig. 2. Architecture details of the implemented network heads. Conv Block means Convolution layer followed by a batch normalization and ReLU activation. Linear Block means Linear layer followed by a batch normalization and ReLU activation

3.2 Estimation of 3D Keypoints

Estimation of 3D points requires the feature maps derived from the image crops initially obtained for the 2D point estimation network and the features extracted from the estimated 2D points by the Keypoint 2D Backbone. For the formation of features from the 2D points, a Multilayer Perceptron (MLP) consisting of seven layers is employed. This MLP takes the normalised 2D coordinates with

respect to the bounding box as input. The feature maps derived from the image and the 2D points are concatenated to create an input for the 3D point estimation head. This module estimates the 3D points in the canonical pose, which is constant, regardless of the observed pose of the vehicle. The canonical pose implies that the coordinate system's origin is situated in the vehicle's geometric center, and the vehicle's front always faces the same direction. This consistent approach to 3D point estimation simplifies the problem, reducing computational complexity while increasing estimation accuracy. The 3D Keypoint Head estimates two separate feature maps. One of these corresponds to the $X-Y$ plane, from which the x and y coordinates are estimated. The second feature map corresponds to the $X-Z$ plane and serves to estimate the z coordinate. By independently addressing two orthogonal planes, this dual-map approach allows us to use a similar processing pipeline that is applied for the estimation of 2D keypoints. The 3D point estimation module's design allows it to seamlessly integrate with the 2D point estimation network by reusing feature maps extracted from the image. Note that our method does not need a mask from pre-trained Mask R-CNN, which is used by the baseline methods considered in [25].

3.3 Selection of the Best Estimated Keypoints

In this section, we introduce a uniquely designed module that focuses on assessing the precision of the point estimations. From any point of view, a significant portion of characteristic points is obscured, making their precise localisation on the image challenging. Such inaccurately estimated points can easily distort pose estimation. To counteract this phenomenon, we have designed an additional head dedicated to evaluating the accuracy of each estimated point. The accuracy evaluation head accepts as input the image feature map, the estimated heatmaps, and the normalised coordinates of the estimated points.

The image feature maps and heatmaps are processed through a sequence of six convolutional layers that outputs two feature vectors, each having length equal to 3072. These feature vectors, along with the estimated point coordinates, are then concatenated and processed through a sequence of six linear layers. The last layer is the sigmoid activation layer, which limits the output values to a range from zero to one. The output of the head is a set of 66 values, each corresponding to the estimation precision score of a given 2D point on the image, hereinafter called KSH scores. During the training phase, as learning targets we used only two values: '1' for a correctly estimated point, and '0' for the opposite case. A point is considered correctly estimated (has a '1' label) if the estimation error normalised to the bounding box is less than 0.04. We also conducted experiments using directly the normalised estimation errors as the learning targets, but the achieved results were worse compared to binary targets generated by thresholding the error.

3.4 Training Process

The training process is organized into distinct stages, each stage focusing on a specific aspect of the network. Initially, during the first stage, we directed our

focus towards the 2D and 3D point heads and the backbone network. Following this, we select the best performing model from this initial training phase. The model's performance is evaluated using the Percentage of Correct Keypoints (PCK) metric for keypoints 2D and Mean Per Joint Position Error (MPJPE) for 3D point estimations. In the PCK metric: $PCK = \frac{P_{correct}}{P} \cdot 100$, where P is the total number of predicted points, we sum up to the number of correct points $P_{correct}$ only the keypoints with the coordinates estimation error normalised to the bounding box smaller than 0.05. The MPJPE metric is defined as a mean of the Euclidean distances between the estimated points and the corresponding ground truth points.

Additionally, for the training of 2D and 3D keypoint heads, we used a reprojection loss function. This function is an extension of an approach presented in [21] that maintains the geometric consistency of the estimated coordinates. It works by comparing the projected 3D points using ground truth transformation with the predictions of the 2D points, ensuring that the network's estimations of 2D and 3D points are consistent with each other and with real-world geometries. This loss is defined as:

$$\mathcal{L}_{\text{reprojection}} = \sum_{i=1}^{n} \left(\left\| \pi \left(\mathbf{T}, \hat{\mathbf{p}}_i^{3d}, \mathbf{K} \right) - \hat{\mathbf{p}}_i^{2d} \right\|_2 \right)^2, \tag{1}$$

where π is the projection function, $\mathbf{T}$ is a ground truth pose, $\hat{\mathbf{p}}_i^{3d}$ are the estimated 3D coordinates of the i-th characteristic point, $\mathbf{K}$ is the camera intrinsics matrix, and $\hat{\mathbf{p}}_i^{2d}$ are the estimated 2D coordinates of the i-th keypoint on image. The application of this loss function significantly improves the accuracy of the point estimation heads.

Once the best model is selected, its weights are frozen to preserve the best results for keypoint estimation. In the second stage, the focus shifts to the training of the Uncertainty Estimation Head and the Keypoint Score Head. By training the UEH and KSH components after the Keypoint 2D and 3D Heads, we ensure that the training process can leverage the well-tuned features provided by the backbone network and the point-generating heads. The loss functions used during training are described by Eq. 2, 3

$$\mathcal{L}_{stage1} = w_{repr} \cdot \mathcal{L}_{reprojection} + \mathcal{L}_{heatmap3Dxy} + \mathcal{L}_{heatmap3Dxz} + \mathcal{L}_{heatmap2D} \tag{2}$$

$$\mathcal{L}_{stage2} = \mathcal{L}_{uncertainty2D} + \mathcal{L}_{uncertainty3D} + \mathcal{L}_{keypoint\_score}, \tag{3}$$

where $w_{repr} = 1e^{-7}$ and $\mathcal{L}_{heatmap3Dxy}, \mathcal{L}_{heatmap3Dxz}, \mathcal{L}_{heatmap2D}, \mathcal{L}_{keypoint\_score}$ are defined as the Mean Squared Error loss function.

3.5 Pose Estimation

Given 2D points in the image, their corresponding 3D points in the model, and the camera matrix $\mathbf{K}$, various PnP algorithms can be applied to derive the pose. The 2D car characteristic points and corresponding 3D points are estimated by our network. During our research, we used two approaches: the Efficient

Perspective-n-Point (EPNP) algorithm [11], and a dedicated procedure called Solve-PnP-BFGS implemented using the SciPy library [28]. For input to the PnP algorithm, we select a subset of N best-estimated points according to KSH scores. In our experiments, we used N=17 as it gives the best results. The Solve-PnP-BFGS procedure works by minimizing the reprojection error using the BFGS optimization algorithm. The reprojection error is defined similarly to (1), but the ground truth transformation $\mathbf{T}$ is replaced by the optimized transformation. The optimisation search space in the BFGS algorithm is constrained by bounds applied to the estimated translation. The optimisation process is carried out five times, with a different starting point randomly selected from the search space each time. The selected solution is the one with the lowest value of the cost function. This repetitive process reduces the possibility of optimisation being stuck at the local minimum. To estimate the shape of a vehicle, we employ a predefined library consisting of all CAD models included in the dataset and a set of 3D characteristic points in canonical form for each model. This library serves as a reference during the shape estimation process, allowing us to capture the full mesh of the vehicles.

During the inference stage, we compare the estimated 3D points with those in the library and select the CAD model that yields the smallest MPJPE. To get the most reliable results, to calculate the MPJPE, we remove firstly the points that are supposed to be the least accurately estimated according to the KSH score value. The best results were achieved by removing points with a KSH score below the threshold $s = 0.19$.

4 Dataset Preparation for Supervised Learning

For our experiments, we use the ApolloCar3D dataset introduced in [25]. This dataset comprises 5,277 images derived from traffic scenes, containing over 60,000 instances of vehicles. Each vehicle instance is defined by a set of 66 characteristic points, with annotations of only visible points. The dataset also provides a set of 34 CAD models of the vehicles appearing in the images. Ground truth pose data, relative to the camera coordinates, are provided for each vehicle. A significant challenge with respect to our pose estimation task is the absence of a mapping between the 2D points in the images and the corresponding 3D points from the CAD models. Such a mapping (provided as labels) is necessary for supervised learning of 3D points detection.

To overcome this problem, we employ a procedure illustrated in Fig. 3. This process begins by transforming the CAD model using the given translation and rotation. Subsequently, the parameters of the ray encompassing all 3D points, which project onto the image at the annotated keypoints for the considered vehicle, are established. For each face of the CAD model, we check if this ray intersects it using the Möller-Trumbore algorithm [19], subsequently determining the coordinates of the intersection point.

In cases where multiple intersection points are discovered, the point closest to the camera is chosen. This strategy is in line with the dataset's approach of marking only non-occluded points. The final stage involves applying the inverse

of the ground truth rotation and translation to the intersection point, yielding the coordinates with respect to the car's canonical pose. The point coordinates derived from this process are then fine-tuned to achieve a more precise match. For this fine-tuning, we employ the Nelder-Mead optimization algorithm on all instances of a particular car type in the training set. This algorithm adjusts the 3D coordinates to minimize the translation error, defined as the square of the distance between the ground truth translation and the translation estimated by the EPnP [11] method, taking into account the given camera parameters and the 2D point coordinates in the image.

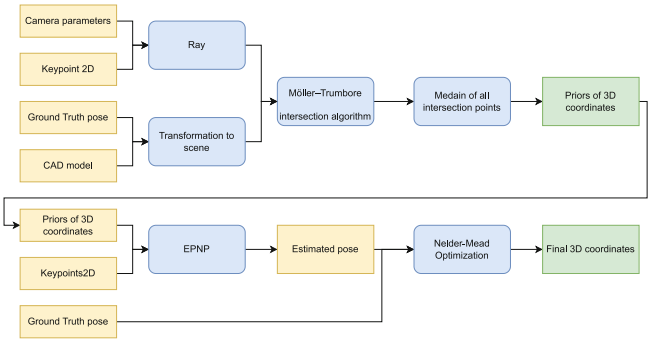

Fig. 3. Pipeline for dataset pre-processing in order to obtain labelled 3D points for supervised learning

5 Uncertainty Estimation

5.1 Estimation of Keypoints Uncertainty

In this section, we discuss a module designed to estimate the uncertainty of characteristic point estimations. The keypoints uncertainty estimation process is an extension of the approach presented in our previous paper [21]. However, our new method estimates both 2D and 3D point uncertainties, contributing to a comprehensive understanding of the estimation process's precision. For each of the 2D points, a 2×2 covariance matrix, $\mathbf{\Sigma}_{2D}$, is estimated for x and y coordinates. The Uncertainty Estimation Head (UEH) consists of a stack of 5 blocks built by convolutional layer, ReLU, and batch normalization. Convolutional blocks are followed by one linear layer. The UEH takes as input the keypoint heatmaps and image feature maps and calculates three positive numbers, filling in the lower triangle of the 2D Cholesky factor matrix, $\mathbf{L}_{2D}$. We ensure the positivity of the estimated numbers through the application of an Exponential Linear Unit (ELU) activation function and the addition of a constant to the output values. Then, by multiplying the matrix $\mathbf{L}_{2D}$ by $\mathbf{L}_{2D}^{T}$, we acquire a covariance matrix $\mathbf{\Sigma}_{2D}$. This approach ensures the generation of a positive semi-definite covariance matrix, a necessary characteristic for valid uncertainty estimation. During the training phase, we employ the Gaussian Log-Likelihood Loss function, as suggested in [8].

When it comes to 3D points, we estimate the values found on the main diagonal of the 3×3 covariance matrix. Only the main diagonal elements are filled in the 3D Cholesky factor matrix, $\mathbf{L}_{3D}$. The subsequent processing steps are analogous to those used in the 2D point uncertainty estimation, reinforcing consistency across the module's operation.

5.2 Propagation Using Unscented Transform

The estimated uncertainties of both 2D and 3D points can be propagated to the uncertainty of the estimated vehicle pose. To accomplish this, we employ the Unscented Transform method, as presented in [22]. The propagation of uncertainty of the n-dimensionsal input is carried out using $2n+1$ sigma points χ_i, $i = 0, ..., n$, while the coordinates of these points are determined by formulas 4:

$$\chi_0 = \mathbf{m_x},\ \chi_{2i-1} = \mathbf{m_x} + \sqrt{n+\lambda}\left[\sqrt{\mathbf{C_x}}\right],\ \chi_{2i} = \mathbf{m_x} - \sqrt{n+\lambda}\left[\sqrt{\mathbf{C_x}}\right], \quad (4)$$

for $i = 1, ..., n$, where $\mathbf{m_x}$ and $\mathbf{C_x}$ are the mean and variance of the estimated points, $\lambda = \alpha^2(n+k) - n$ is a scaling factor, and α and k are the parameters influencing how far the sigma points are away from the mean. Then, by applying a nonlinear transformation, which is the PnP algorithm, we obtain a set of points from which the mean and covariance of the estimation after the nonlinear transformation are estimated. In our implementation, the Unscented Transform parameters α and k [22] are set to 0.9 and 50, respectively.

The propagation process for 2D and 3D points is executed independently. After the uncertainties have been individually propagated, the resulting covariance matrices of equal dimension are summed up, assuming statistical independence between the coordinates of the 3D and 2D keypoints.

6 Evaluation Results

Model evaluation was carried out on the ApoloCar3D dataset validation set. It contains 200 images according to the split made by the authors of the ApoloCar3D dataset. On a Nvidia 1080Ti GPU the proposed pipeline is capable to run at 20 FPS for the EPNP variant without uncertainty propagation and at 18 FPS with full uncertainty propagation.

6.1 Keypoints 2D

For the evaluation of the 2D keypoints estimation network, we used the Percentage of Correct Keypoints metric, assuming three thresholds: 5 pixels, 10 pixels, and 15 pixels.

The first two rows in Table 1 were calculated using as reference only visible vehicle points, i.e., those located in the ground truth annotations of the dataset. We provide metric values for all marked points and for those points that the network identified as having the highest KSH score. The results demonstrated that the network correctly assesses the accuracy of the point estimation and is capable of improving the results achieved on the PCK metric.

The last two rows in Table 1 present analogous PCK metric values. The difference is that we used the projections of 3D points as ground truth annotations, using the ground truth translation and rotation of a given vehicle. This approach allows for an evaluation of the invisible points that is more relevant to real-world conditions, where we do not have a priori knowledge about the visibilities of points. The presented results, especially those acquired for the points selected by the KSH score provide sufficient accuracy for the 3D pose estimation using PnP algorithms.

Table 1. PCK metric of the 2D keypoints estimations comparing to manually labeled visible points (top) and all 66 points acquired by projection of 3D points (bottom)

Threshold	5 px	10 px	15 px
Visible points PCK	55.0	81.0	89.0
Selected visible points PCK	60.0	84.0	91.0
All points PCK	13.1	19.7	22.2
Selected points PCK	40.5	56.0	60.9

6.2 Keypoints 3D

In this section, we present the results related to the estimation of 3D points on vehicles using our proposed method. Module performance is evaluated using the Mean Per Joint Position Error (MPJPE) metric, which measures the average distance error of the estimated points. Our results reveal an averaged MPJPE value of 0.119 m for all points estimated by the network. When we consider only the N points with the highest KSH score, the error decreases to 0.105 m, which confirms that the KSH score promotes better keypoints. The most accurately estimated points are on the rear corner of the car handle of the right front car door, with a mean error of 0.073 m, while the points of lowest accuracy (0.235 m on average) are located on the left corner of the rear bumper.

6.3 A3DP Metrics

We utilised the Absolute 3D Pose Error (A3DP-Abs) metric introduced in [25] to evaluate our results. This metric focuses on the absolute distances to objects, considering three key components: the estimated shape of the car, its position, and its rotation. The translation error metric is defined as:

$$c_{\text{trans}} = \left|\mathbf{t}_{gt} - \hat{\mathbf{t}}\right|_2 \leq \delta_t, \tag{5}$$

where $\mathbf{t}_{gt}$ denotes ground truth translation, $\hat{\mathbf{t}}$ denotes estimated translation and δ_t is an acceptance threshold. The rotation error metric is defined as:

$$c_{\text{rot}} = \arccos\left(|\mathbf{q}_{gt} \cdot \hat{\mathbf{q}}|\right) \leq \delta_{rot}, \tag{6}$$

where $\mathbf{q}_{gt}$ denotes ground truth rotation quaternion, $\hat{\mathbf{q}}$ denotes estimated rotation quaternion, and δ_{rot} is an acceptance threshold. Similarly to metrics proposed for the COCO dataset, the authors of ApolloCar3D proposed a set of

metric thresholds from strict to loose. The translation thresholds were established from 2.8 m to 0.1 m in increments of 0.3 m, while the rotation thresholds range from $\pi/6$ to $\pi/60$ with steps of $\pi/60$. Beyond the 'mean' metric, which averages results across all thresholds, two single-threshold metrics were defined. The loose criterion (denoted as $c-l$) utilises [2.8, $\pi/6$] thresholds for translation and rotation, whereas the strict criterion (denoted as $c-s$) employs [1.4, $\pi/12$] as thresholds for these respective parameters. To evaluate the 3D shape reconstruction, a predicted mesh is rendered from 100 different perspectives. Intersection over Union (IoU) is computed between these renderings and the ground truth masks. The average of these IoU calculations is then used to evaluate the shape reconstruction.

Table 2 shows our results and compares them against state-of-the-art methods. Note that for the algorithms proposed in [2,9] the implementations provided in [25] as baselines were used for a fair comparison.

Table 2. Comparison of results with the state-of-the-art methods on A3DP-Abs metrics

algorithm	mean	$c-l$	$c-s$
3D-RCNN [9]	16.4	29.7	19.8
DeepMANTA [2]	20.1	30.7	23.8
GSNet [7]	18.9	37.4	18.4
BAAM-Res2Net [10]	25.2	47.3	23.1
Ours EPnP	23.4	44.6	31.7
Ours BFGS	**25.6**	**47.7**	**34.6**

The results of the A3DP-Abs mean metric indicate that our system's performance is superior to the very recent BAAM [10] method, which demonstrates the proficiency of our solution in estimating 3D points accurately. A significant advantage of our network is its performance on the strict criterion $c-s$. This metric measures how well our module estimates 3D points under stringent conditions, a challenging task that demands a high degree of precision and reliability. Our system outperforms all existing state-of-the-art solutions on this metric by a large margin, highlighting the solution's excellence in providing highly accurate 3D characteristic points.

6.4 Uncertainty Assessment

In this section, we present results related to the uncertainty estimation of the 2D and 3D characteristic points, as well as to the uncertainty of the vehicle's pose. The first part of our evaluation process involved examining the percentage of point and vehicle translation estimations that fall within the ranges of 1, 2, and 3 standard deviations (σ). The results of this examination are shown in the

Table 3. Percentage of estimation that fall within the ranges of one, two and three σ

	Keypoints 2D		Keypoints 3D			Pose translation		
	x	y	x	y	z	x	y	z
1 σ	79.0	79.8	82.5	80.3	64.8	84.5	85.0	81.3
2 σ	93.1	93.8	94.2	94.1	80.9	94.9	95.6	93.7
3 σ	97.2	97.2	97.5	97.3	87.5	98.6	98.6	97.5

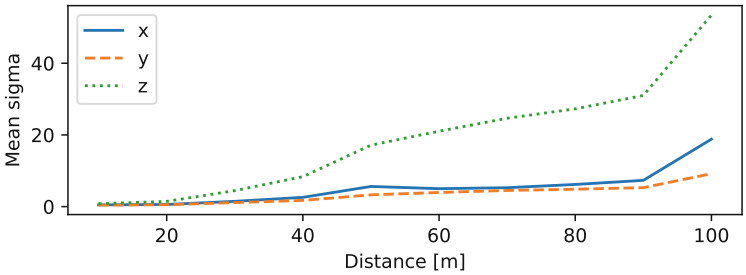

Fig. 4. Plot of the mean σ (standard deviation) depending on the observation distance

Table 3. We further examine the relationship between the mean value of σ and the vehicle's distance in Fig. 4.

These findings suggest that as distance increases, the uncertainty in pose estimation also increases. This trend is to be expected, as distant objects inherently have the lower resolution in the image, which tends to lead to higher uncertainties in the estimation. An additional noteworthy observation is that the uncertainty along the z-axis (depth direction) is greater than along the remaining two axes. This finding is justified, as it is often more challenging to determine the position along the axis perpendicular to the image plane. Due to the nature of monocular vision and image projection, depth information is less reliable and can result in higher uncertainties. Figure 5 presents six examples of predictions and the uncertainty of the estimated pose from a bird's eye view. The uncertainty of position (x, y) for each vehicle is represented by its uncertainty ellipse. Observe that cars that are partially occluded or are located further from the camera have larger uncertainty ellipses than the closer, fully visible cars.

Fig. 5. Visualizations of estimated pose and uncertainty. Circles show the estimated positions of cars, squares are ground truth positions of these cars, colour rays show ground truth yaw angles, grey fans show the estimated one-sigma yaw range. Ellipses visualise one sigma position uncertainty (Color figure online)

7 Conclusions

In this paper, we have introduced a comprehensive processing pipeline capable of estimating vehicle pose solely based on data from a single RGB camera. This innovative pipeline comprises a network for estimating 2D and 3D points, an

evaluation of the credibility of individual point estimations, and an optimisation algorithm estimating pose based on information derived from previous steps.

Our system outperforms all previously reported models on the ApolloCar3D dataset and is capable of estimating the uncertainty of pose estimates. The unique ability to handle uncertainty has significant implications for real-world applications, offering an additional layer of validation and error mitigation in pose estimation and further decision making.

We also introduce an unsupervised method for preparing a training dataset containing 3D point coordinates. This approach leverages 2D point annotations, CAD models, and vehicle poses, offering a novel methodology to generate reliable training data without the need for laborious and error-prone manual annotation.

In conclusion, our approach represents a significant advancement in the field of vehicle pose estimation, providing a powerful, robust, and accurate pipeline for this complex task. Furthermore, the ability to estimate the uncertainty of pose estimations and the introduction of an unsupervised method for training dataset preparation pave the way for future advancements in this field. Further work will include the integration of geometric priors stemming from the scene structure and semantics, such as the coplanarity constraint applied in [25].

References

1. Brazil, G., Liu, X.: M3D-RPN: monocular 3D region proposal network for object detection. In: IEEE/CVF International Conference on Computer Vision (ICCV), pp. 9286–9295 (2019)
2. Chabot, F., Chaouch, M., Rabarisoa, J., Teuliere, C., Chateau, T.: Deep MANTA: A coarse-to-fine many-task network for joint 2D and 3D vehicle analysis from monocular image. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2040–2049 (2017)
3. Ferraz, L., Binefa, X., Moreno-Noguer, F.: Leveraging feature uncertainty in the PNP problem. In: Proceedings of the British Machine Vision Conference (2014)
4. Haralick, R.M.: Propagating covariance in computer vision. Int. J. Pattern Recogn. Artif. Intell. **10**(5), 561–572 (1996)
5. Hoque, S., Xu, S., Maiti, A., Wei, Y., Arafat, M.Y.: Deep learning for 6d pose estimation of objects - a case study for autonomous driving. Expert Syst. Appl. **223**, 119838 (2023)
6. Huang, J., Zhu, Z., Guo, F.: The devil is in the details: delving into unbiased data processing for human pose estimation. arXiv:2008.07139 (2020)
7. Ke, L., Li, S., Sun, Y., Tai, Y.-W., Tang, C.-K.: GSNet: joint vehicle pose and shape reconstruction with geometrical and scene-aware supervision. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12360, pp. 515–532. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58555-6_31
8. Kumar, A., Marks, T.K., Mou, W., Feng, C., Liu, X.: UGLLI face alignment: Estimating uncertainty with gaussian log-likelihood loss. In: IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), pp. 778–782 (2019)
9. Kundu, A., Li, Y., Rehg, J.M.: 3D-RCNN: instance-level 3D object reconstruction via render-and-compare. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3559–3568 (2018)

10. Lee, H.J., Kim, H., Choi, S.M., Jeong, S.G., Koh, Y.J.: BAAM: monocular 3d pose and shape reconstruction with bi-contextual attention module and attention-guided modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9011–9020, June 2023
11. Lepetit, V., Moreno-Noguer, F., Fua, P.: EPnP: an accurate O(n) solution to the PnP problem. Int. J. Comput. Vision **81**, 155–166 (2009)
12. Li, B., Zhang, T., Xia, T.: Vehicle detection from 3d lidar using fully convolutional network. In: Hsu, D., Amato, N.M., Berman, S., Jacobs, S.A. (eds.) Robotics: Science and Systems XII. University of Michigan, Ann Arbor (2016)
13. Li, H., et al.: Pose-oriented transformer with uncertainty-guided refinement for 2d-to-3d human pose estimation (2023)
14. Li, P., Chen, X., Shen, S.: Stereo R-CNN based 3D object detection for autonomous driving. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7636–7644 (2019)
15. Li, P., Zhao, H., Liu, P., Cao, F.: RTM3D: Real-time monocular 3d detection from object keypoints for autonomous driving. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12348, pp. 644–660. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58580-8_38
16. Liu, F., Hu, Y., Salzmann, M.: Linear-covariance loss for end-to-end learning of 6d pose estimation. CoRR abs/2303.11516 (2023)
17. LÃşpez, J.G., Agudo, A., Moreno-Noguer, F.: Vehicle pose estimation via regression of semantic points of interest. In: 11th International Symposium on Image and Signal Processing and Analysis (ISPA), pp. 209–214 (2019)
18. Marti, E., de Miguel, M.A., Garcia, F., Perez, J.: A review of sensor technologies for perception in automated driving. IEEE Intell. Transp. Syst. Mag. **11**(4), 94–108 (2019)
19. Möller, T., Trumbore, B.: Fast, minimum storage ray-triangle intersection. J. Graph. Tools **2**(1), 21–28 (1997)
20. Mousavian, A., Anguelov, D., Flynn, J., Kosecka, J.: 3d bounding box estimation using deep learning and geometry. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR. pp. 5632–5640 (2017)
21. Nowak, T., Skrzypczyński, P.: Geometry-aware keypoint network: accurate prediction of point features in challenging scenario. In: 17th Conference on Computer Science and Intelligence Systems (FedCSIS), pp. 191–200 (2022)
22. PÃl'rez, D.A., Gietler, H., Zangl, H.: Automatic uncertainty propagation based on the unscented transform. In: IEEE International Instrumentation and Measurement Technology Conference (I2MTC), pp. 1–6 (2020)
23. Reddy, N.D., Vo, M., Narasimhan, S.G.: Occlusion-Net: 2D/3D occluded keypoint localization using graph networks. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7318–7327 (2019)
24. Shi, J., Yang, H., Carlone, L.: Optimal pose and shape estimation for category-level 3d object perception. arXiv:2104.08383 (2021)
25. Song, X., et al.: ApolloCar3D: a large 3D car instance understanding benchmark for autonomous driving. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5447–5457 (2019)
26. Toshpulatov, M., Lee, W., Lee, S., Haghighian Roudsari, A.: Human pose, hand and mesh estimation using deep learning: a survey. J. Supercomput. **78**(6), 7616–7654 (2022)
27. Vakhitov, A., Colomina, L.F., Agudo, A., Moreno-Noguer, F.: Uncertainty-aware camera pose estimation from points and lines. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4657–4666 (2021)

28. Virtanen, P., et al.: SciPy 1.0 contributors: SciPy 1.0: fundamental algorithms for scientific computing in python. Nat. Methods **17**, 261–272 (2020). https://doi.org/10.1038/s41592-019-0686-2
29. Wang, J., et al.: Deep high-resolution representation learning for visual recognition. IEEE Trans. Pattern Anal. Mach. Intell. **43**, 3349–3364 (2021)
30. Wang, Q., Chen, J., Deng, J., Zhang, X.: 3D-CenterNet: 3D object detection network for point clouds with center estimation priority. Pattern Recogn. **115**, 107884 (2021)
31. Xu, B., Chen, Z.: Multi-level fusion based 3D object detection from monocular images. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2345–2353 (2018)
32. Yang, B., Luo, W., Urtasun, R.: PIXOR: real-time 3D object detection from point clouds. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7652–7660 (2018)
33. Yang, H., Pavone, M.: Object pose estimation with statistical guarantees: conformal keypoint detection and geometric uncertainty propagation. CoRR abs/2303.12246 (2023)