Proceedings of the of the 17th Conference on Computer Science and Intelligence Systems pp. 191–200
DOI: 10.15439/2022F145
ISSN 2300-5963 ACSIS, Vol. 30

Geometry-Aware Keypoint Network: Accurate Prediction of Point Features in Challenging Scenario

Tomasz Nowak
Institute of Robotics and Machine Intelligence
Poznań University of Technology
ul. Piotrowo 3A, 60-965 Poznań, Poland
Email: tomasz.nowak@doctorate.put.poznan.pl

Piotr Skrzypczyński
Institute of Robotics and Machine Intelligence
Poznań University of Technology
ul. Piotrowo 3A, 60-965 Poznań, Poland
Email: piotr.skrzypczynski@put.poznan.pl

***Abstract*—In this paper, we consider a challenging scenario of localising a camera with respect to a charging station for electric buses. In this application, we face a number of problems, including a substantial scale change as the bus approaches the station, and the need to detect keypoints on a weakly textured object in a wide range of lighting and weather conditions. Therefore, we use a deep convolutional neural network to detect the features, while retaining a conventional procedure for pose estimation with 2D-to-3D associations. We leverage here the backbone of HRNet, a state-of-the-art network used for detection of feature points in human pose recognition, and we further improve the solution adding constraints that stem from the known scene geometry. We incorporate the reprojection-based geometric priors in a novel loss function for HRNet training and use the object geometry to construct sanity checks in post-processing. Moreover, we demonstrate that our Geometry-Aware Keypoint Network yields feasible estimates of the geometric uncertainty of point features. The proposed architecture and solutions are tested on a large dataset of images and trajectories collected with a real city bus and charging station under varying environmental conditions.**

Fig. 1. Accurate detection of keypoint features for vision-based localisation of an electric bus while docking to a charging station. The inset images show a charging station with the ground truth keypoints (upper), the estimated locations of these keypoints with uncertainty ellipses (lower), and a close-up of the roof-mounted sensor unit with a camera (in the circle)

I. Introduction

Estimation of the absolute pose of a camera from a single image is a classic problem in computer vision [1], which is being solved in a number of ways, depending on the application. Conventional, structure-based algorithms yield accurate camera pose estimates if stable, salient point features can be extracted. Unfortunately, in a number of practical scenarios, particularly outdoors, good quality point features (keypoints) are hard to extract from images, due to weak textures, difficult lighting, and motion blur.

The localisation scenario considered in this paper is part of the Advanced Driver Assistance System (ADAS) which we developed in cooperation with Solaris Bus & Coach, one of the manufacturers of electric buses [2]. The ADAS provides a bus driver with visual cues on how to operate the steering wheel in order to perform the desired manoeuvre. To perform successful docking, the driver has to put the tip of the vehicle's pantograph into the head of the charger, which is mounted on a support pylon. An important prerequisite for successfully planning such a manoeuvre is an accurate estimate of the bus position in relation to the charger head and its pylon. The use of GPS for localising while docking is considered unreliable in an urban environment. Active beacons or even large passive markers installed at the charging station cannot be considered as well, because bus operators often do not permit deployment of any additional elements at the stations due to legal issues. For localisation, the bus has only a monocular camera mounted in the front part of the roof. The use of such a simple sensor was required by the bus manufacturer, as the ADAS equipment has to be affordable and scalable to different bus models.

The charging station is detected automatically from a long distance (typically 30 m), using the approach introduced in [3]. Once the station gets detected, the task comes down to the estimation of the camera pose with respect to certain predefined points of the charger structure. Despite its apparent simplicity, this scenario raises a number of issues in the camera pose estimation method. Firstly, the method needs to work without an initial pose guess, considering each observation of the charging station as a separate global localisation act. Secondly, the charging station is assumed to be the only known object in the environment, for which we have a 3-D model. Hence, we need to use the keypoints defined on the charging station that are observed during the entire docking manoeuvre over a large range of viewpoints, appearance, and scale change. All these difficulties make the existing simultaneous localisation

and mapping (SLAM) or visual odometry (VO) algorithms impractical in our scenario, as SLAM and VO systems need to be initialised with a known camera pose and require the salient features to be present in the environment all along the executed trajectory.

To address the specific requirements we proposed in [4] a two-step procedure, which uses a conventional, structure-based pose estimation algorithm with 2D-to-3D associations between the keypoints found in the camera image and predefined points from the three-dimensional model of the charger. Whereas this procedure using a Faster R-CNN network architecture adopted to detect the keypoints has been positively verified in docking experiments with a real bus [2], [4], we observed, that any inaccuracy in the position of the estimated keypoints significantly deteriorates the accuracy of the final position estimate. A mismatch of detected points with other features makes the pose estimate completely wrong.

Therefore, in this paper we investigate how to adapt to our application a different network architecture: the High Resolution Network (HRNet) [5], which is a leading solution for keypoint detection in human pose estimation. The detection of body points in humans is a major area of interest for researchers and commercial use, and thus it arguably sets the state-of-the-art in keypoint detection in the wild [6]. We conjecture that a network architecture achieving top scores in the COCO Keypoint Detection Task would be a good starting point for use in the considered application. However, our application offers some a priori knowledge about the geometry of the observed object, which is not present in the human pose estimation task. Thus, we investigate how to include this knowledge either in the learning process, creating an inductive bias in the neural network, or in post-processing, defining a sanity check procedure that quickly eliminates implausible predictions of the network. We also extend the neural network architecture by adding a separate branch that estimates covariance matrices describing the 2-D uncertainty of the detected keypoints. Finally, we get the new Geometry-Aware Keypoint Network that addresses the specific challenges of the bus localisation process at the charging station.

In this work, we contribute: (i) an analysis of the HRNet architecture aimed at the accuracy improvement of the keypoints locations with respect to ground truth, also considering the computation burden; (ii) a novel loss function that exploits the available geometric priors of our application; (iii) a sanity check procedure based on these geometric priors, and (iv) an extended experimental evaluation of all these components on a unique application-specific dataset, which is made publicly available.

The remainder of this paper is organised as follows. Section II reviews the most important related work in similar applications and in keypoints detection from monocular images. Section III gives a description of the proposed neural network architecture, and provides technical details pertaining to its novel components. Then, section IV describes the experimental procedure and summarises the results of the experiments, while section V draws conclusions upon these results.

II. Related work

A. Vision for automated docking

There are few works concerning vision-guided docking of larger vehicles to electric chargers [7]. Precise docking to a charging station under the control of a camera can be cast as a visual servoing problem. Unfortunately, visual servoing methods [8] require the target object or marker to appear big enough in the images, whereas the charging station detected from a distance of 30 m is too small to make a visual servoing method effective. Recent visual SLAM algorithms can estimate a camera's trajectory precisely over hundreds of metres [9]. In practice, however, a SLAM method requires a large number of features detected over several consecutive image frames [10] and has to be initialised properly, which is problematic in the monocular case and often requires several attempts with camera relocation. These problems make SLAM impractical for the specific task we consider, as neither visual odometry nor SLAM exploit the knowledge about the appearance and geometry of the charging station, which is assumed to be visible during the entire docking manoeuvre.

On the other hand, automated charging for self-driving electric cars is often implemented using devices that plug into the car's charging port [11]. This approach eases the guidance process, as only this port has to be localised with respect to the plugging arm/device. In this context, the approach we propose is of practical importance, as we do not need any active devices, nor markers/fiducials attached to the charging station, and we can localise the bus during the entire manoeuvre, starting 30 metres from the station.

B. Camera pose estimation

A wide variety of algorithms have been presented in the literature for the camera pose estimation problem. The classic way is to compute the camera pose from 2D-to-3D correspondences between 2-D features in the image and 3-D points in the model [1]. The model can be given a priori, from CAD data or via accurate laser scanning, as in our scenario, or can be obtained through 3-D structure-from-motion (SfM). Correspondences between points are established through the matching of descriptors, which can either be handcrafted [12] or learned [13]. The camera pose is then computed upon the known correspondences applying a variant of the perspective-n-point (PnP) algorithm [14] or optimised using bundle adjustment [15]. The conventional approach can estimate the camera pose accurately [16], but it struggles whenever the features are difficult to match. Our approach follows the classic pipeline with respect to pose computation using a PnP algorithm, but deploys a specialised neural network as the feature detector, thus obtaining a well-defined pattern of a few keypoints that are already associated with the model points. Owing to this concept we do not need to use RANSAC for outlier rejection.

In the last few years, learning-based approaches to camera pose estimation have been gaining attention. End-to-end methods have been pionieered by PoseNet [17], which used a trained convolutional neural network to regress a six degrees-of-freedom camera pose. A similar idea was followed by

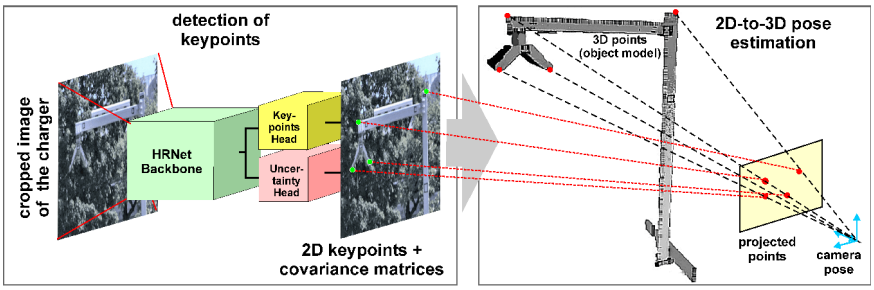

Fig. 2. Structure of the camera pose estimation system consisting of a CNN-based keypoint detection module and a conventional pose computation method

[18], demonstrating camera relocalisation in cluttered scenes. Although PoseNet can localise the camera without an initial pose guess, as we require in our scenario, the accuracy of pose estimates yielded by this neural network is inferior with respect to its conventional, structure-based counterparts using principled algorithms. As concluded in [19], the main reason for the insufficient accuracy was the use of a naive regression loss function without consideration of the scene geometry. The improved PoseNet presented in [19] uses reprojection-based loss, thus narrowing the performance gap to pose estimation using principled algorithms. The importance of using the geometric knowledge about the scene and enforcing the geometric constraints during learning was pointed out in [20], and has been demonstrated recently also by other pose estimation methods. A camera pose estimation pipeline that is applicable for fusing classical geometry and deep learning is proposed in [21], while [22] introduces an end-to-end system consisting of deep learning modules for feature extraction, matching, and outlier rejection while optimising for the camera pose objective.

C. Detection of keypoints

In conventional, structure-based pose estimation systems detectors and descriptors designed by heuristics are applied. However, in recent years, deep learning has been applied to obtain feature detectors and descriptors in visual odometry and SLAM. LIFT [23] was among the first attempts to detect features by end-to-end learning, and was trained on ground truth obtained from SIFT and SfM. The more recent SuperPoint [13] introduced a self-supervised approach to learn the detectors and descriptors of keypoints simultaneously, while DISK [24] demonstrated learning of features using policy gradient.

Independently, a number of neural network architectures for keypoint detection have been proposed in the context of human pose estimation. The architecture from [25] is based on Faster R-CNN with ResNet-101 backbone, similarly to our previous camera pose estimation system described in [4]. Whereas this approach yields accurate keypoints, and thus is included in our experiments for comparison, it is computationally heavy and hence does not allow to exploit the full resolution of the acquired images. A recent paper [26] demonstrated the use of novel self-calibrated convolutions that expand fields-of-view of each convolutional layer to detect keypoints. A leading solution for keypoint detection in human pose estimation is the High Resolution Network [5], which was designed specifically to keep the high resolution of the feature maps through the entire processing pipeline. This approach is in line with our idea of using high-resolution images for reliable detection of keypoints from longer distances, while still keeping good accuracy of their localisation in images. Therefore, we selected the HRNet as our baseline approach and adopted its backbone network as a feature extractor in our GAKN architecture.

Uncertainty in learned feature extractors and camera pose estimation methods seems to be not yet fully exploited, despite its importance in the conventional, geometric approaches. Among the examples considering uncertainty, [27] uses Bayesian neural networks to obtain localisation uncertainty, and a recent approach [21] learns the deep neural network uncertainty guided by the geometric uncertainty. In our localisation scenario, we are interested in aleatoric uncertainty, which depends on the inputs, and may be estimated from data [28]. Whereas Bayesian deep learning is popular for this purpose, we leverage the Cholesky Estimator Network to represent the uncertainty of each keypoint as a Gaussian distribution with covariance matrix [29].

III. Structure of the Proposed Solution

The main inspiration for this work was an excellent performance of state-of-the-art keypoint detectors in the top down human pose estimation methods [6]. In the top down approach to human pose estimation keypoint locations are predicted within bounding boxes obtained from a person detector. This fits our processing pipeline, where the charging station is detected by a Faster R-CNN object detector [3], and further processing for localisation is limited to the image area cropped by the object's bounding box [4] (Fig. 2).

Therefore, we adopt the keypoint detection architecture with HRNet [5] backbone implemented using the MMPose framework [30] as a baseline model during our experiments. This architecture consists of a backbone network and the keypoint head. We decided to use a pre-trained backbone, and design our own head, also adding an auxiliary branch for the estimation of spatial uncertainty of the keypoints (Fig. 3). The keypoint head contains Deconv Blocks which double the resolution of the feature maps. A single Deconv Block is built with a Transposed Convolution Layer followed by a Batch

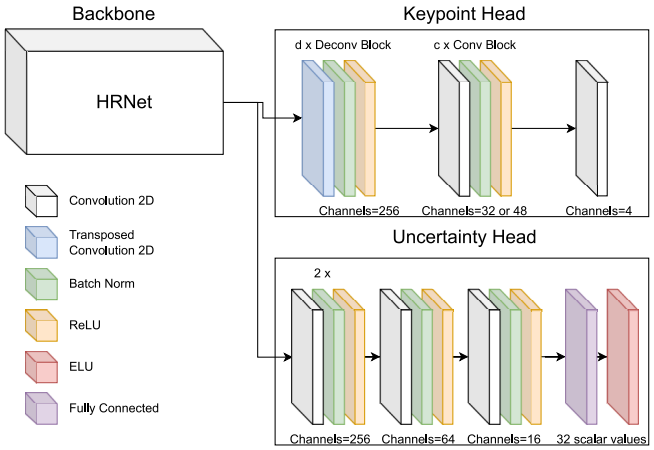

Fig. 3. Architecture of the GAKN model. Configurations with heatmap size of 128×128, 256×256 and 512×512 contains d equal 0, 1 or 2 Deconv Blocks respectively. Architectures can be extended with c additional Conv Blocks in the keypoint head

Norm and ReLU layers. The final layer in the keypoint head is a single Convolution Layer that outputs n heatmaps, where n is the number of keypoints. For the preparation of ground truth heatmaps as training targets and the decoding of the final keypoint coordinates from the heatmaps we incorporated the UDP and DarkPose methods described in [31] and [32], respectively. These data preprocessing techniques allow us to preserve the accuracy of ground truth coordinates of the keypoints through the labelling and augmentation process. During the inference, while decoding the coordinates from predicted heatmaps, those methods allow for extracting unbiased, subpixel-accuracy keypoint locations.

Point detection in objects such as an electric bus charger is different from body point detection in humans. The human body is composed of numerous parts moving relatively to each other, therefore choosing which part of the image should be used to look for a given point is not obvious. Due to the rigid geometric configuration of the charging station, this is not a problem in our network design, thus we focused on developing methods to increase the accuracy of point localisation. We experimented with several configurations of the HRNet architecture in order to find the limits of the keypoints location accuracy, and to choose a reasonable trade-off between the accuracy and the number of computations. We selected three aspects of the HRNet model for potential improvements: the selection of a backbone network for feature extraction, the size of the returned heatmaps, and the structure of the keypoint head. Next, having a reasonable baseline design customised to our application, we investigated the best way of injecting the prior geometric knowledge to the network model.

A. Backbone network

We evaluated two backbone networks HRNet32 and HRNet48. The difference between HRNet48 and HRNet32 lies in the width of the convolutional layers in the high-resolution stream (48 and 32 respectively). This enhances the capability of HRNet48 to extract more feature maps but at the cost of higher computational cost.

B. Keypoint head depth

We evaluated the influence of the additional convolutional layers in the keypoint head to find out whether it will improve the pose estimation accuracy. The extra layers were added after the Deconv Block and before the convolution layer which produces final heatmaps. Each extra layer consists of 256 filters of the size 3×3 and is followed by Batch Norm and ReLU.

C. Heatmap size

The default implementation of the keypoint detector based on HRNet returns heatmaps which are downsampled four times compared to the input image size, so using an image of 512×512 pixels results in 128×128 pixels heatmaps. The upsampling of the heatmaps is achieved using Transposed Convolutional layers. A single transposed convolutional layer increases the width and height of the heatmap twice.

The charging station pylon and head do not contain moving parts and their geometric configuration is constant. This makes rough estimation of the keypoints locations easier than the same task in human pose estimation because we can expect the given point in the specific area of the image. On the other hand, the accuracy of the described pose estimation method is strongly dependent on the accuracy of estimation of the keypoints. Small inaccuracies in the location of keypoints propagate to relatively large pose estimation errors, especially from larger distances. The above considerations suggest, that increasing the resolution of the output heatmaps may promote accurate subpixel estimation of the keypoint locations, which in turn will lead to a significant improvement of the camera pose estimation accuracy.

D. Reprojection loss

The key role of scene geometry in conventional pose estimation models and the fact that ground truth points can be easily identified for the charging station, together with the geometric relations between these points, were inspirations for developing the HRNet baseline model into the Geometry-Aware Keypoint Network.

This network exploits these geometric priors while training, as it uses an additional cost function based on the reprojection error. The reprojection loss penalises spatial configurations of the keypoints which are physically impossible. We define camera projection function, π, which maps the i-th 3-D point $\mathbf{w}_i$ to the 2-D image point $(\tilde{u}, \tilde{v})^T$ leveraging the given camera intrinsics parameters $\mathbf{K}$:

$$\pi(\mathbf{T}, \mathbf{K}, \mathbf{w}_i) \mapsto (\tilde{u}, \tilde{v})^T, \tag{1}$$

where $\mathbf{T}$ is a rigid transformation matrix (rotation and translation).

To calculate the reprojection loss, we minimise the difference between the projection of the real 3-D object points $(\tilde{u}, \tilde{v})^T$ and the points predicted by the network$(\hat{u}, \hat{v})^T$. The optimisation problem is solved by the Trust Region Reflective algorithm [33] which finds a transformation $\mathbf{T}^*$:

$$\mathbf{T}^* = \underset{\mathbf{T}}{\operatorname{argmin}} \sum_{i=1}^{n} \left(\tilde{u_i} - \hat{u_i}\right)^2 + \left(\tilde{v_i} - \hat{v_i}\right)^2 . \tag{2}$$

Trust Region Reflective is a robust optimisation method for constrained problems that has a Python implementation in the SciPy library, which facilitates integration in a deep learning framework.

To limit the search space and keep the solution physically correct we applied constraints on the transformation $\mathbf{T}$ to reflect only the operating area of the bus. All three values of the rotation vector are limited to $\frac{\pi}{4}$. Assuming that roll and pitch angle are close to zero, this constraint limits the yaw angle to be less than $\pm\frac{\pi}{4}$ The translation in the lateral axis is limited to $\pm$ 20 m, the longitudinal axis is limited to 50 m and the translation in the z axis is limited to 50 m (Fig. 4).

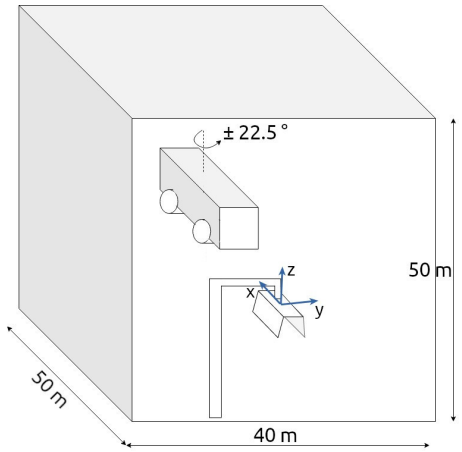

Fig. 4. The considered search space for the bus location and orientation.

Then, the reprojection loss is the value of the cost function for the optimal $\mathbf{T}^*$ transformation:

$$\text{loss}_{\text{repr}} = \sum_{i=1}^{n} \left(\pi(\mathbf{T}^*, \mathbf{K}, w_i^x) - \hat{u_i}\right)^2 + \left(\pi(\mathbf{T}^*, \mathbf{K}, w_i^y) - \hat{v_i}\right)^2 , \tag{3}$$

where w_i^x and w_i^y are x and y coordinates of point $\mathbf{w}_i$.

The second loss element is a Mean Squared Error Loss:

$$\text{loss}_{\text{MSE}} = \sum_{i=1}^{n} \left(\frac{1}{m} \sum_{j=1}^{m} \left(m_{ij} - \hat{m_{ij}}\right)^2 \right), \tag{4}$$

where m_{ij} is the j-th pixel of the ground truth heatmap corresponding to the i-th point and $\hat{m_{ij}}$ is the j-th pixel of the predicted heatmap corresponding to the i-th point. The final loss function is a sum of the MSE loss (4) and the reprojection loss, calculated according to the formula:

$$\text{loss} = \text{loss}_{\text{MSE}} + \lambda \cdot \text{loss}_{\text{repr}}, \tag{5}$$

where the metaparameter λ was estimated experimentally and is set to 0.01 for our evaluation.

E. Reprojection-based refinement

Each of the four points defined on the charging station can be detected more or less accurately, depending on a number of circumstances, including the camera viewpoint and the amount of motion blur in the image. In this situation, it may happen, that some keypoints are well extracted, while some others are inaccurate or even misplaced.

To cope with this problem we define a sanity check procedure that incorporates the geometric constraints during the inference of the neural network. This procedure refines the neural network predictions, solving the same task as described by the equation (2). Having the optimal transformation $\mathbf{T}^*$, we project 3-D coordinates of the keypoints on the image. Then the distances between the predicted point $(\hat{u}, \hat{v})^T$ and the projected point $(\tilde{u}, \tilde{v})^T$ are calculated. Then, we compare the maximum distance d_{max} with the mean of three remaining distances d_{res} multiplied by parameter $\gamma = 2$. If the inequality $d_{max} > \gamma d_{mean}$ is satisfied, we use this projection as the final prediction of keypoints location. The above condition ensures that we are refining only the cases where there is only one point with inaccurate prediction.

F. Uncertainty estimation

As knowing the geometric uncertainty of extracted keypoints may be instrumental when computing the camera pose estimate and then fusing it with other localisation data in the vehicle, we extended the GAKN architecture with an uncertainty estimation branch. Our aim was to obtain a trainable model that predicts covariance matrices of the individual keypoints depending on the input images. With such a model we can judge if the extracted keypoints are accurate enough for the camera (and then vehicle) localisation task. For this purpose, we implemented the Gaussian Log-Likelihood Loss proposed in [29]. We estimate the uncertainty of all four keypoint locations as a Gaussian distribution with covariance matrix $\mathbf{\Sigma}$, an 8×8 symmetric positive definite matrix. $\mathbf{\Sigma}$ has $(2n+1)n$ degrees of freedom which are estimated by a lower triangular matrix $\mathbf{L}$ such that $\mathbf{\Sigma} = \mathbf{L}\mathbf{L}^\mathbf{T}$ (Cholesky decomposition). The GAKN uncertainty estimation branch consists of four blocks built with a convolutional layer followed by Batch Norm and ReLU. The final layer is Fully Connected which predicts 36 values of the $\mathbf{L}$ matrix. The loss function used during training is described by:

$$\text{loss}_{\text{unc}} = \sum_{i=1}^{n} \log|\mathbf{\Sigma}| + (\mathbf{p} - \hat{\mathbf{c}})\, \mathbf{\Sigma}^{-\mathbf{1}}\, (\mathbf{p} - \hat{\mathbf{c}}), \tag{6}$$

where $\mathbf{p}$ is a vector of ground truth keypoint locations and $\hat{\mathbf{c}}$ is a vector of predicted keypoint locations. The 2×2 covariance matrices of the individual keypoints are then extracted from $\mathbf{\Sigma}$, and can be visualised as uncertainty ellipses in the image plane.

IV. Experiments

A. Evaluation procedure

The purpose of the presented experiments was to examine the influence of the neural network architecture on the 2-D pose estimation accuracy and the computational complexity. All experiments were performed off-line using Nvidia A100 GPU on a custom dataset recorded using a real electric bus and charger with the ground truth poses obtained using Differential GPS (DGPS). In our experimental installation (cf. Fig. 1) a high-resolution camera (5472×3648 pixels) is used, as the application scenario requires to detect the charging station from a distance of at least 30 m and localise with a 2-D circular position error smaller than 0.35 m when approaching the charger's head (this error can be compensated mechanically). The camera is mounted in a detachable unit with a laser scanner for safety monitoring (not used here), and a DGPS receiver, which was used only to obtain ground truth trajectories of the bus. Note that compared to [4] we no longer use the black rectangular markers that are visible on the charger's pylon (cf. Fig. 1). The ground truth keypoints are defined by manual labelling directly on the training sequence images. They are two corners of the charger's head and two other points on the extreme parts of the supporting pylon. The dataset consists of 81 image sequences gathered over five days. The diversity of data was achieved by different manoeuvre starting points, different bus trajectories, and changing lighting and weather conditions.

The final pose estimate of the camera is computed as in [4], using an iterative perspective-n-point algorithm, which minimises the reprojection error. The cost function of this optimisation problem is defined as sum of squared distances between the point localisation on the image, and the object model points projected on this image:

$$\mathbf{T}^* = \underset{\mathbf{T}}{\operatorname{argmin}} \sum_{i=1}^{n} \left((\hat{u}, \hat{v})_i^T, \pi\left(\mathbf{T}\mathbf{w}_i\right)\right)^T \left((\hat{u}, \hat{v})_i^T, \pi\left(\mathbf{T}\mathbf{w}_i\right)\right). \tag{7}$$

To consider the detection of a keypoint as accepted, the RMSE of the 3-D point projected on the image using ground truth transformation from DGPS should be less than d pixels:

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\sqrt{(\tilde{u}-\hat{u})^2+(\tilde{v}-\hat{v})^2}\right)^2} < d, \tag{8}$$

with d set to 10 in our experiments.

Finally, we evaluated the performance of the different neural models applied for keypoints extraction directly for the objective of camera pose accuracy. The ground truth trajectories were provided by DGPS (2 cm of accuracy), while the camera and DGPS receiver were calibrated using an optimisation-based procedure developed specifically for this project [34]. The 3-D model of the charging station was obtained using a geodetic laser scanning system. For evaluation we use three metrics: median of the 2-D translation error, the median of the yaw angle estimation, and percentage of accepted detections. We project the 3-D camera pose into the 2-D manifold and ignore the pitch and roll angles, as these dimensions are irrelevant to localisation of the bus on a flat area. For all experiments, the image input size was set to 512×512 pixels. Results of the main experiments are presented in Tab. I. The considered configurations of the keypoint extraction network are the Baseline (i.e. HRNet32 with 3 layers head), Refined, which is Baseline with the geometric sanity check, GAKN, and GAKN+Refined, which is GAKN equipped with the sanity check in the post-processing.

B. Comparison with the state-of-the-art

To compare the GAKN architecture with the state-of-the-art we use the HRNet configured for the considered application scenario according to the outcome of our experiments, i.e. with the HRNet32 backbone and the keypoint head with 3 layers, which we call the "Baseline" configuration. Additionally, we use for comparison the Faster R-CNN-based network described in [25]. During the inference, this network required above 32 GB of GPU memory, most of it was consumed by the extraction of the keypoint location offset values. The inference time is significantly longer and the pose estimation errors are larger than for the HRNet-based networks. In our application scenario, the weakest point of the network from [25] is a low percentage of accepted detections (40.9 %). To improve this parameter, we tested also our sanity check procedure (post-processing) with this network architecture. However, in this case, application of the reprojection-based refinement does not bring the expected results. The ratio of accepted detections increased to above 50 % but the pose estimation accuracy dropped. This case shows that the reprojection-based refinement can improve results only when the raw predictions are relatively close to ground-truth and there is usually only a single keypoint of lower accuracy.

C. Backbone network

Comparing the HRNet32 and the HRNet48 (Tab. II) we can notice that the medians of rotation and translation errors are lower for the HRNet48 network for both keypoint head variants. There is no significant influence on the percentage of accepted detections. The inference time is slightly longer for the HRNet48 version and the required operations and number of parameters are over twice as large as for the HRNet32 backbone. Using the HRNet48-based network, the whole processing pipeline requires more than 8 GB of GPU memory. The commercial application of this system requires that the cost of used hardware must be considered. Despite better pose estimation accuracy, we selected the HRNet32 backbone for the Baseline, as it provides acceptable accuracy and can be run on low-end hardware (GPU). Consequently, the GAKN architecture configurations evaluated in our experiments also use the HRNet32 backbone.

D. Keypoint head depth

Additional convolutional layers in the keypoint head decrease slightly the median of translation and rotation error (Tab. II). Applying this modification does not affect the

TABLE I
COMPARISON OF THE SIZE OF NETWORKS, INFERENCE TIME, POSE ESTIMATION ERRORS AND PERCENT OF ACCEPTED DETECTIONS OF THE EVALUATED ARCHITECTURES.

Heatmap size	Configuration	Operations [GFLOPs]	Parameters [M]	Inference time [ms]	Median t_2D [m]	Median r_2D [deg]	Percent of accepted detections
152×152	Papandreou	41.22	42.67	167.82	0.53	1.15	40.90 %
	Papandreou+Refined			171.82	1.16	1.67	52.34 %
128×128	Baseline	41.08	28.54	**35.60**	0.43	0.97	92.59 %
	Refined			40.60	0.44	0.98	95.66 %
	GAKN			**35.60**	0.37	0.67	92.14 %
	GAKN+Refined			40.60	0.37	0.67	94.79 %
256×256	Baseline	43.34	28.67	38.40	0.35	0.74	93.56 %
	Refined			42.40	0.36	0.74	95.99 %
	GAKN			38.40	0.32	0.62	95.56 %
	GAKN+Refined			42.40	0.32	**0.61**	**96.60** %
512×512	Baseline	112.47	29.72	50.60	0.32	0.70	94.03 %
	Refined			54.60	0.32	0.71	95.02 %
	GAKN			50.60	**0.30**	0.64	92.82 %
	GAKN+Refined			54.60	0.31	0.64	94.44 %

TABLE II
COMPARISON OF THE SIZE OF NETWORKS, INFERENCE TIME, POSE ESTIMATION ERRORS AND PERCENT OF ACCEPTED DETECTIONS OF THE NETWORKS WITH DIFFERENT BACKBONES AND DEPTHS OF THE KEYPOINT HEAD.

Heatmap size	Backbone	Convolution layers in the keypoint head	Operations [GFLOPs]	Parameters [M]	Inference time [ms]	Median t_2D [m]	Median r_2D [deg]	Percent of accepted detections
128×128	HRNet32	1	41.08	28.54 M	**35.40**	0.46	0.98	93.59 %
	HRNet48		84.10	63.60 M	37.10	0.42	0.72	92.83 %
	HRNet32	3	41.12	28.54 M	35.60	0.43	0.97	**93.89 %**
	HRNet48		84.18	63.60 M	37.60	**0.40**	**0.71**	92.87 %

percentage of accepted detections and has a marginal influence on the number of operations and inference time.

E. Heatmap size

In this subsection, we compare the results of the Baseline network on three different heatmap resolutions. The default implementation of the keypoint detector based on HRNet returns heatmaps which are downsampled four times compared to the input image size, so using an image of 512×512 pixels results in 128×128 pixels heatmaps. The influence on the percentage of accepted detections is marginal, but increasing the heatmap size significantly reduces the translation and rotation error. The difference in the number of operations between a 128×128 and a 256×256 heatmap is small, compared to the 512×512 version, where the number of operations increases three times. This translates into processing time for a single image, wherein the 512×512 version is characterised by the processing time twice as large as in the 128×128 version. Increasing the size of the processed heatmap allows for increased location accuracy, but at the cost of increased processing time. However, there is only a slight increase in the number of network parameters when increasing the heatmap size. This means that all of the architectures discussed, including the largest one, should fit on a graphics card with 8 GB of memory. As the size of the heatmap increased, we achieved a reduction in errors (e.g., the translation error of a 128-sized heatmap relative to a 512-sized heatmap was 26.7%, the rotation error in the same ratio was 33.9%), which was the main research goal. When considering the errors of a 256-sized heatmap, they were larger than 512 and smaller than a 128-sized heatmap confirming the relationship between heatmap size and location accuracy. Heatmap size does not affect the percentage of accepted detections.

F. Reprojection loss

The configuration of the GAKN network features the same number of operations, parameters, and processing times because the only modification to the baseline network was made during the training phase and does not affect these parameters during inference. There is a reduction in translation and rotation errors in all three heatmap sizes considered. When using reprojection loss for heatmap size 128, there was a 15.2 % reduction in translation error, 37 % in rotation error, for size 256, the translation error decreased by 8.4 %, and rotation error decreased by 15.5 %, whereas for heatmap size 512: there was a 2.6 % reduction in translation error and 9.5 % in rotation error. By combining both approaches - increasing the heatmap size and applying reprojection loss, we obtain the lowest median translation error among all tested models, and a median rotation error comparable to the lowest obtained with heatmap size 256. We believe that using a heatmap size of 256 is a reasonable trade-off between processing time and location accuracy. It achieved the best rotation error result, translation error comparable to the best result, and features a short inference time – similar to the one for 128×128 heatmap. Additionally, the percentage of accepted detections in the GAKN configuration with this heatmap size is higher than in the baseline.

TABLE III
PERCENTAGE OF GROUND-TRUTHS WITHIN UNCERTAINTY ELLIPSE FOR DIFFERENT STANDARD DEVIATIONS

Standard deviation	1σ	2σ	3σ
Percentage of ground-truths within uncertainty ellipse	48.75 %	87.38 %	98.00 %

G. Reprojection-based refinement

Using reprojection-based refinement does not affect the number of operations performed by the network and the number of parameters, because it is performed at the post-processing stage. On average, this step takes 4 ms per image, which constitutes 11% of the processing time for heatmap 128×128, and only 8 % for heatmap 512×512, therefore adding reprojection-based refinement does not increase significantly the image processing time. No clear effect on location accuracy was observed when using reprojection-based refinement. The main goal of this method was to increase the percentage of accepted detections, which was achieved. As a result, we are able to provide a higher number of correct pose estimates, making the localisation system more reliable. By using the GAKN network together with reprojection-based refinement, we eliminate the relatively low percentage of accepted detections.

H. Uncertainty estimation

Using a hand-labeled validation dataset of about 200 images, we evaluated the geometric uncertainty prediction. We checked the percentage of ground truth keypoint locations that are within the 1, 2, and 3 σ uncertainty ellipses. The numerical values are presented in Tab. III.

Qualitative results of covariance matrices prediction are demonstrated in Fig. 5. Comparing Fig. 5A and Fig. 5B, we can notice that the uncertainty of keypoint detection decreases with the distance to the charger. Figure 5C and Fig. 5E show larger uncertainty for poor quality images. On Fig. 5D, the two leftmost points have larger uncertainty along x axis because, from that point of view, estimation of their exact location is ambiguous.

I. Analysis of cumulative distribution plots

Comparing the 3 pairs of graphs (Fig. 6, Fig. 7 and Fig. 8) corresponding to heatmap sizes, it can be seen that the benefit of reprojection loss is biggest for heatmap size 128×128, and decreases as the heatmap size increases. It can also be seen that by using reprojection-based refinement, we increase the number of accepted detections that provide coarse location information, despite the relatively large translation and rotation error (right part of the graph). Furthermore, we are able to improve the detection accuracy for which the error is below the median (the red line in Fig. 6. is above the green line for values below 0.4 m and 0.6 deg).

J. Network attention analysis

Starting this research we conjectured that the prior geometric knowledge represented by the reprojection-based loss component introduces to the GAKN architecture an inductive bias that helps the network to focus on the most relevant image areas when searching for the keypoints. To verify this claim, we implemented in GAKN an attention analysis layer, based on the Score-CAM method [35]. Compared to the older, gradient-based methods Score-CAM produces results that are visually less noisy, making it easier to interpret the attention depending on the input image.

In Fig. 9 and Fig. 10, the top row (A, B, C and D) shows the activation maps for all 4 detected points in an example image from the test set. Comparing the activation maps shown, it can

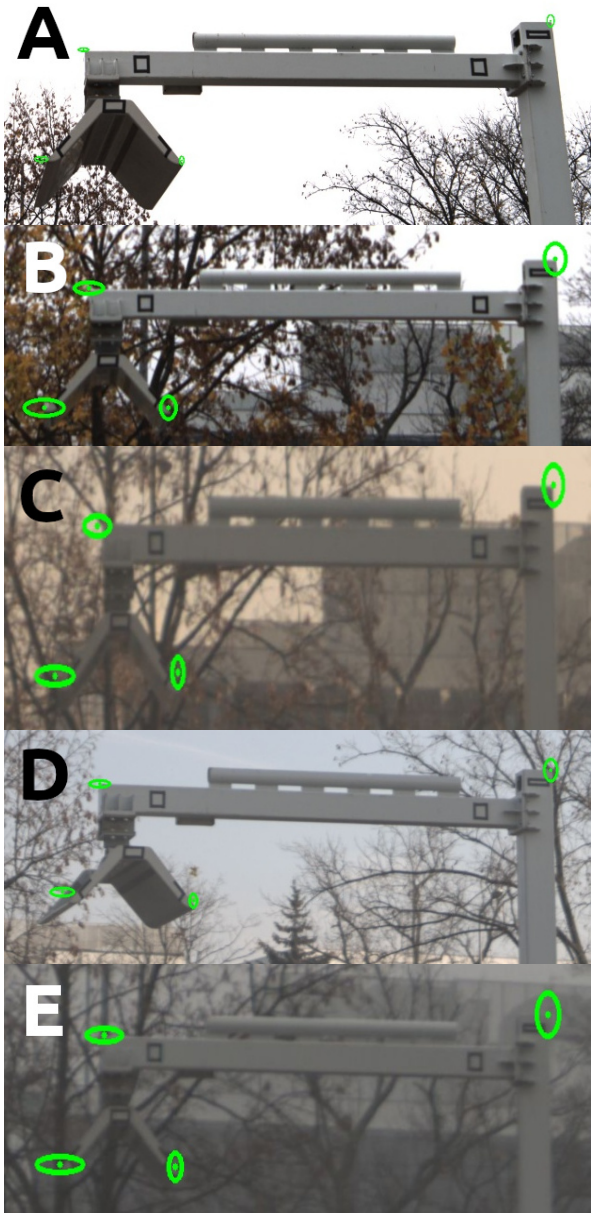

Fig. 5. Visualisation of estimated keypoints locations and 3 sigma uncertainty ellipses for different observation cases. Fig. A presents detection from close distance, B - far distance, C - blurry image, D - different observation angle, E - cloudy/foggy weather

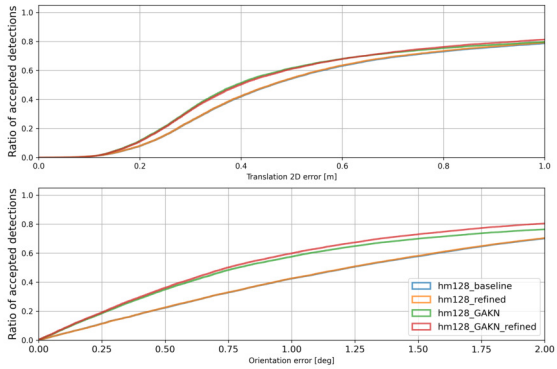

Fig. 6. Cumulative distribution functions of 2D translation error (A) and orientation error (B) for the HRNet32 models with the heatmap size 128×128

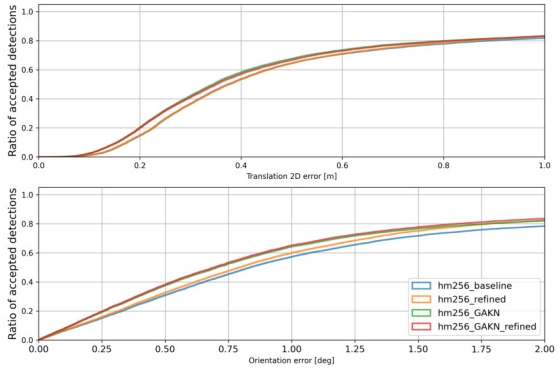

Fig. 7. Cumulative distribution functions of 2D translation error (A) and orientation error (B) for the HRNet32 models with the heatmap size 256×256

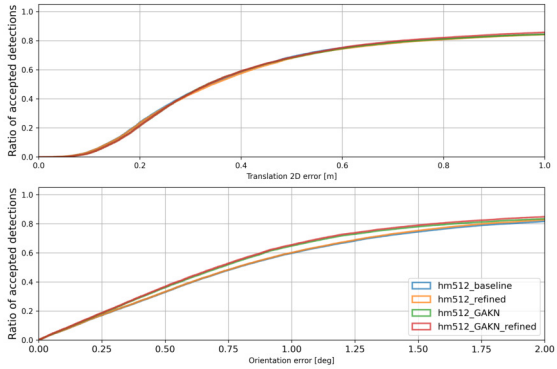

Fig. 8. Cumulative distribution functions of 2D translation error (A) and orientation error (B) for the HRNet32 models with the heatmap size 512×512

be seen that the activations of individual points for the GAKN network are more concentrated and have higher intensity. In the bottom row of Fig. 9 and Fig. 10 (E, F, G, and H), the Score-CAM method marked the parts of the image that had

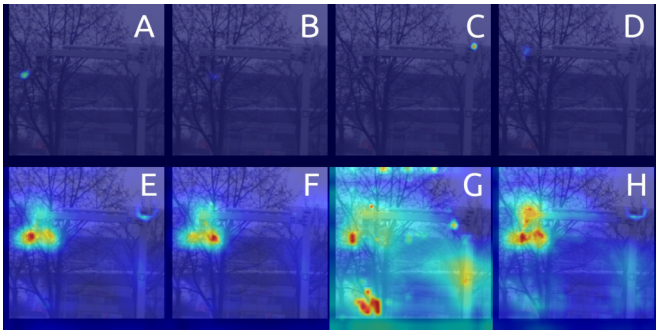

Fig. 9. Heatmaps (top) and the output of the Score-CAM algorithm (bottom) from the baseline HRNet32. Warmer colors mean higher activation

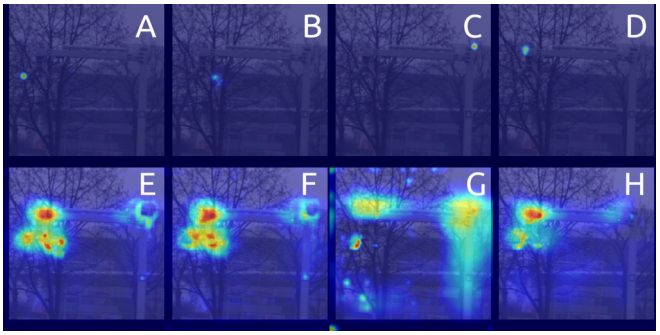

Fig. 10. Heatmaps (top) and the output of the Score-CAM algorithm (bottom) from the Geometric Aware Keypoint Network. Warmer colors mean higher activation

the greatest effect on selecting a specific location in the image as the searched point. The first thing you notice is that in the case of the baseline network, activation is scattered throughout the background image, indicating that the network attention is highly scattered. Furthermore, in the case of point three (G), there is very weak activation near the actual point location and large activation on a portion of the image completely unrelated to the charger structure (lower left corner). In Fig. 10, showing points from the GAKN network, the network's attention is more focused near the searched points than at the baseline.

V. Conclusion

This paper evaluated our experience with applying the state-of-the-art HRNet architecture for the detection of keypoints in a challenging scenario of camera pose estimation for localisation of an electric bus. The proposed GAKN model achieved better results than both the Faster R-CNN-based keypoint detector and the baseline HRNet32. Our solution takes less than 50 ms for processing a single image on Nvidia A100, which makes it possible to update the camera pose frequently and facilitates accurate bus localisation along its path to the charging station. Reliable localisation under changing viewpoint, scale, lighting, and weather conditions is achieved without any markers deployed at the charging station. The contributed modifications to the baseline HRNet, based on exploiting the available geometric priors were evaluated positively, as they improve the accuracy of keypoint locations.

Finally, we evaluated the novel elements of GAKN with respect to the localisation accuracy objective, and we have found that using the reprojection loss reduces translation and rotation errors, while using refinement (sanity check) in postprocessing increases the number of accepted detections, thus increasing the availability of the camera pose estimates. Our future research will determine if the proposed GAKN architecture can be scaled down to edge computing platforms, for better cost-efficiency. The project with datasets is available at https://github.com/ZephyrII/mmpose_charger

ACKNOWLEDGMENT

This work is partially under the project "Advanced driver assistance system (ADAS) for precision maneuvers with single-body and articulated urban buses", co-financed within the Smart Growth Operational Programme 2014-2020 (POIR.04.01.02-00-0081/17-01). P. Skrzypczyński is supported by TAILOR, a project funded by EU Horizon 2020 under GA No. 952215. T. Nowak is supported by PUT internal grant 0214/SBAD/0235.

REFERENCES

[1] R. I. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge University Press, 2004.

[2] M. M. Michalek, T. Gawron, M. Nowicki, and P. Skrzypczynski, "Precise docking at charging stations for large-capacity vehicles: An advanced driver-assistance system for drivers of electric urban buses," *IEEE Vehicular Technology Magazine*, vol. 16, no. 3, pp. 57–65, 2021.

[3] T. Nowak, M. Nowicki, K. Ćwian, and P. Skrzypczyński, "How to improve object detection in a driver assistance system applying explainable deep learning," in *IEEE Intelligent Vehicles Symposium*, Paris, 2019, pp. 226–231.

[4] ——, "Leveraging object recognition in reliable vehicle localization from monocular images," in *Automation 2020: Towards Industry of the Future*, ser. AISC, vol. 1140. Cham: Springer, 2020, pp. 195–205.

[5] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, W. Liu, and B. Xiao, "Deep high-resolution representation learning for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, pp. 3349–3364, 2021.

[6] M. Toshpulatov, W. Lee, S. Lee, and A. Haghighian Roudsari, "Human pose, hand and mesh estimation using deep learning: a survey," *The Journal of Supercomputing*, vol. 78, no. 6, pp. 7616–7654, 2022.

[7] L. G. Clarembaux, J. Pérez, D. Gonzalez, and F. Nashashibi, "Perception and control strategies for autonomous docking for electric freight vehicles," *Transportation Research Procedia*, vol. 14, pp. 1516–1522, 2016, transport Research Arena TRA2016.

[8] E. Marchand, F. Spindler, and F. Chaumette, "ViSP for visual servoing: a generic software platform with a wide class of robot control skills," *IEEE Robotics and Automation Magazine*, pp. 40–52, 2005.

[9] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: an accurate open-source library for visual, visual-inertial, and multimap SLAM," *IEEE Trans. Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[10] K. L. Lim and T. Bräunl, "A review of visual odometry methods and its applications for autonomous driving," *arXiv*, vol. 2009.09193, 2020.

[11] J. Miseikis, M. Ruther, B. Walzel, M. Hirz, and H. Brunner, "3d vision guided robotic charging station for electric and plug-in hybrid vehicles," *arXiv*, vol. 1703.05381, 2017.

[12] J. Ma, X. Jiang, A. Fan, J. Jiang, and J. Yan, "Image matching from handcrafted to deep features: A survey," *International Journal of Computer Vision*, vol. 129, no. 1, pp. 23–79, 2021.

[13] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint: Self-supervised interest point detection and description," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2018*, 2018, pp. 224–236.

[14] V. Lepetit, F. Moreno-Noguer, and P. Fua, "EPnP: An accurate o(n) solution to the pnp problem," *International Journal of Computer Vision*, vol. 81, no. 2, 2008.

[15] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment — a modern synthesis," in *Vision Algorithms: Theory and Practice*. Berlin, Heidelberg: Springer, 2000, pp. 298–372.

[16] T. Sattler, C. Sweeney, and M. Pollefeys, "On sampling focal length values to solve the absolute pose problem," in *Computer Vision – ECCV 2014*. Cham: Springer, 2014, pp. 828–843.

[17] A. Kendall, M. Grimes, and R. Cipolla, "Posenet: A convolutional network for real-time 6-dof camera relocalization," in *IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 2938–2946.

[18] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "PoseCNN: a convolutional neural network for 6D object pose estimation in cluttered scenes," in *Proceedings of Robotics: Science and Systems*, Pittsburgh, Pennsylvania, June 2018.

[19] A. Kendall and R. Cipolla, "Geometric loss functions for camera pose regression with deep learning," in *IEEE Conference on Computer Vision and Pattern Recognition, (CVPR)*, 2017, pp. 6555–6564.

[20] T. Sattler, Q. Zhou, M. Pollefeys, and L. Leal-Taixé, "Understanding the limitations of cnn-based absolute camera pose regression," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 3297–3307.

[21] B. Zhuang and M. Chandraker, "Fusing the old with the new: Learning relative camera pose with geometry-guided uncertainty," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 32–42.

[22] Y.-Y. Jau, R. Zhu, H. Su, and M. Chandraker, "Deep keypoint-based camera pose estimation with geometric constraints," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 4950–4957.

[23] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, "LIFT: learned invariant feature transform," in *Computer Vision - ECCV 2016 - 14th European Conference, Proceedings, Part VI*, ser. LNCS, vol. 9910. Springer, 2016, pp. 467–483.

[24] M. J. Tyszkiewicz, P. Fua, and E. Trulls, "DISK: learning local features with policy gradient," in *Advances in Neural Information Processing Systems 33 (NeurIPS)*, 2020.

[25] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. P. Murphy, "Towards accurate multi-person pose estimation in the wild," *arXiv*, vol. 1701.01779, 2017.

[26] J.-J. Liu, Q. Hou, M.-M. Cheng, C. Wang, and J. Feng, "Improving convolutional networks with self-calibrated convolutions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[27] A. Kendall and R. Cipolla, "Modelling uncertainty in deep learning for camera relocalization," in *IEEE International Conference on Robotics and Automation, (ICRA)*, 2016, pp. 4762–4769.

[28] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?" *arXiv*, vol. 1703.04977, 2017.

[29] A. Kumar, T. K. Marks, W. Mou, C. Feng, and X. Liu, "Uglli face alignment: Estimating uncertainty with gaussian log-likelihood loss," in *IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 2019, pp. 778–782.

[30] MMPose, "Openmmlab pose estimation toolbox and benchmark," https://github.com/open-mmlab/mmpose, 2020.

[31] J. Huang, Z. Zhu, and F. Guo, "The devil is in the details: Delving into unbiased data processing for human pose estimation," *arXiv*, vol. 2008.07139, 2020.

[32] F. Zhang, X. Zhu, H. Dai, M. Ye, and C. Zhu, "Distribution-aware coordinate representation for human pose estimation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[33] M. Branch, T. Coleman, and Y. Li, "A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems," *SIAM J. Sci. Comput.*, vol. 21, pp. 1–23, 1999.

[34] M. R. Nowicki, "A data-driven and application-aware approach to sensory system calibration in an autonomous vehicle," *Measurement*, vol. 194, p. 111002, 2022.

[35] H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, and X. Hu, "Score-CAM: score-weighted visual explanations for convolutional neural networks," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2020.