ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-6327-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICASSP49357.2023.10096758

ROBUST MONOCULAR LOCALIZATION OF DRONES BY ADAPTING DOMAIN MAPS TO DEPTH PREDICTION INACCURACIES

Priyesh Shukla, Sureshkumar S., Alex C. Stutts, Sathya Ravi, Theja Tulabandhula, and Amit R. Trivedi

University of Illinois at Chicago (UIC), Chicago, USA

ABSTRACT

We present a novel monocular localization framework by jointly training deep learning-based depth prediction and Bayesian filtering-based pose reasoning. The proposed cross-modal framework significantly outperforms deep learning-only predictions with respect to model scalability and tolerance to environmental variations. Specifically, we show little-to-no degradation of pose accuracy even with extremely poor depth estimates from a lightweight depth predictor. Our framework also maintains high pose accuracy in extreme lighting variations compared to standard deep learning, even without explicit domain adaptation. By openly representing the map and intermediate feature maps (such as depth estimates), our framework also allows for faster updates and reusing intermediate predictions for other tasks, such as obstacle avoidance, resulting in much higher resource efficiency.

Index Terms— Depth neural network, drone localization.

1 Introduction

For self-navigation, the most fundamental computation required for a vehicle is to determine its position and orientation, i.e., *pose* during motion. Higher-level path planning objectives such as motion tracking and obstacle avoidance operate by continuously estimating vehicle's pose. Recently, deep neural networks (DNNs) have shown a remarkable ability for vision-based pose estimation in highly complex and cluttered environments [1–3]. For visual pose estimation, DNNs can learn the correlation of vehicle's position/orientation and visual fields to a mounted camera. Thereby, vehicle's pose can be predicted using a monocular camera alone. In contrast, the traditional methods required bulky and power-hungry range sensors or stereo vision sensors to resolve the ambiguity between an object's distance and its scale [4, 5].

However, DNN's *implicit learning* of flying domain features such as its map, placement of objects, coordinate frame, domain structure, *etc.* in a standard pose-DNN also affects the robustness and adaptability of pose estimations. The traditional filtering-based approaches [6] account for the flying space structure using explicit representations such as voxel grids, occupancy grid, Gaussian mixture model (GMM), *etc.* [7]; thereby, updates to the flying space such as map extension, new objects, and locations can be more easily accommodated. Comparatively, DNN-based estimators cannot handle selective map updates, and the entire model must be retrained even under small randomized or structured perturbations. Additionally, filtering loops in traditional methods can adjudicate predictive uncertainties against measurements to systematically prune hypothesis space and can express prediction confidence along with the prediction itself [8]. Whereas feed-forward pose estimations from a deterministic DNN are vulnerable to measurement and modeling uncertainties.

In this paper, we use integrate traditional filtering techniques with deep learning to overcome such limitations of DNN-based pose estimation while exploiting their suitability to operate efficiently with monocular cameras alone. Specifically, we present a novel framework for visual localization by integrating DNN-based depth prediction and Bayesian filtering-based pose localization. In Fig. 1, avoiding range sensors for localization, we utilize a DNN-based lightweight depth prediction network at the front end and sequential Bayesian estimation at the back end. Our key observation is that, unlike pose estimation, which innately depends on map characteristics such as spatial structure, objects, coordinate frame, *etc.*, depth prediction is *map-independent* [9, 10]. Thus, by applying deep learning only on domain-independent tasks and utilizing traditional models where domain is openly (or explicitly) represented helps improve the predictive robustness. Limiting deep learning to only domain-independent tasks also allows our framework to utilize vast training sets from unrelated domains. Open representation of map and depth estimates enables faster domain-specific updates and utilization of intermediate feature maps for other autonomy objectives, such as obstacle avoidance.

2 Monocular Localization with Depth Neural Network and Pose Filters

In Fig. 1, our framework integrates deep learning-based depth prediction and Bayesian filters for visual pose localization in the 3D space. At the front end, a depth DNN scans monocular camera images to predict the relative depth of image pixels from the camera's focal point. A particle filter localizes the camera pose at the back end by evaluating the likelihood of

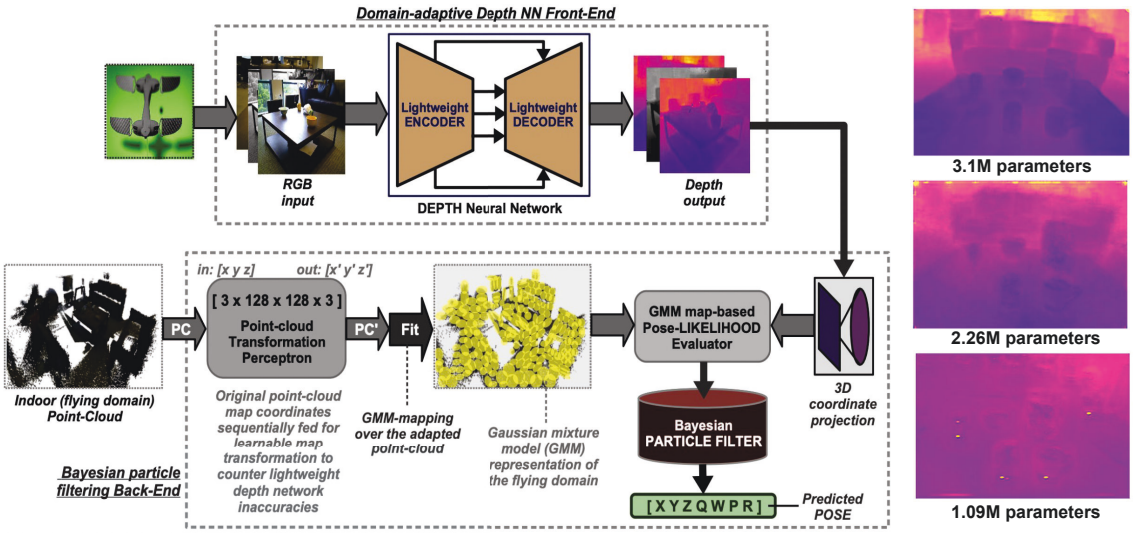

Fig. 1: Proposed framework integrating depth estimator front-end and particle filter back-end for extremely lightweight and robust localization. DNN-based preprocessing avoids area/power-hungry range sensors. The filtered response of the back-end predictor is robust against measurement and modeling uncertainties. On the right are depth predictions from a lightweight network with varying model sizes.

3D projection of depth scans over a GMM-based map representation of 3D space. Both frameworks are jointly trained for the extremely lightweight operation. Various components of the framework are discussed below:

2.1 Extremely Lightweight Depth Prediction

DNN-based monocular depth estimation has gained wide interest owing to impressive results. Several fully supervised [11], self-supervised [12], and semi-supervised [13] convolutional neural network (CNN)-based depth estimators have been presented with promising results. However, for low-power edge robotics [14], the existing depth DNNs are often oversized. A typical depth DNN combines an encoder that extracts the relevant features from the input images. The features are then up-sampled using a decoder to predict the depth map. *Skip connections* between various encoding and decoding layers are typically used to obtain high-resolution image features within the encoder which in-turn helps the decoding layers reconstruct a high resolution depth output.

In Fig. 2, we consider a depth DNN that integrates state-of-the-art architectures for lightweight processing on mobile devices. The depth predictor uses MobileNet-v2 as encoder and RefineNet [15] as decoder. MobileNet-v2 concatenates memory-efficient inverted residual blocks (IRBs). The intermediate layer outputs (or RGB-image features) from the encoder are decoded through the successive channels of convolutional sum, chained residual pooling (CRP), and depth-feature upsampling. This architecture uniquely utilizes only 1×1 convolutional layers in SUM and CRP blocks (replacing traditional high receptive field 3×3 CONV layers with 1×1 CONV layers), thus significantly reducing model parameters. Due to the modular architecture of the depth predictor in Fig. 2, its size can be scaled down by reducing the number of layers in the encoder and decoder. However, with fewer parameters, the prediction quality is affected. Fig. 1 (on the right) shows the depth quality by reducing the number of model parameters. Later, we will discuss how despite lower quality depth prediction, accurate pose localization can be achieved by adapting maps to depth inaccuracies.

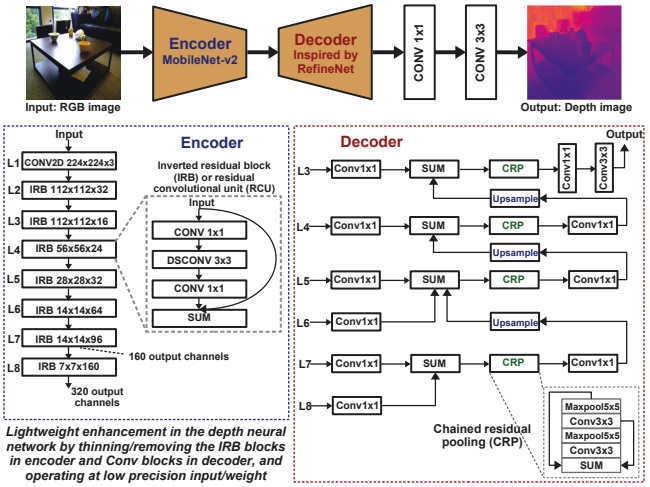

Fig. 2: Depth neural network architecture with MobileNet-v2 [16] encoder and decoder based on RefineNet [15].

2.2 Memory Efficient Mapping using GMMs

To minimize the memory footprint of maps, we utilized a GMM-based representation of 3D maps [17]. The point-cloud distribution of tested maps was clustered and fitted with a 3D GMM using Expectation-Maximization (EM) procedures. Although alternate map representations are prevalent, the parametric formulation of GMMs can considerably minimize the necessary storage and extraction cost. For ex-

ample, Voxel grids [18] use cells to represent dense maps and are simpler to extract. However, the representation suffers from storage inefficiency since the free space also needs to be encoded. Surfels, 3D planes, and triangular mesh [19] are storage efficient, however expensive to retrieve map information from. Generative map modeling using GMMs requires only the storage of the means, variances, and mixing proportions. GMM-maps easily adapts to scene complexity, that is, for more complex scenes, we can use more mixture components as necessary.

2.3 Adapting Maps to Depth Mispredictions

In Fig. 2, lightweight depth network with fewer parameters or layers induces significant inaccuracies in the predicted depth map. Therefore, the accuracy of pose estimation suffers. We discuss integrated learning of depth and pose reasoning to overcome such deficiencies of lightweight predictor.

In Fig. 1, we integrate a multi-layer perceptron (MLP)-based learnable transform (size: $3\times128\times3$) to the original point-cloud (PC) map that minimizes the impact of lightweight depth predictor by translating and/or rotating map points adaptively to systematic inaccuracies of the predictor. The last layer of the depth predictor is also tuned. A joint training of map transformations and depth predictor is quite expensive since each update iteration involves nested sampling and particle filtering steps. The complexity of parameter filtering can be significantly minimized using techniques such as hierarchical GMM representations [7], beat-tracking [20], *etc.*, however, the resultant formulation is non-differentiable, precluding gradient descent-based optimization.

To circumvent the training complexity, instead of directly minimizing ℓ_2 norm of the predicted and ground truth pose trajectory, we minimize the negative log-likelihood (NLL) of input image projection via lightweight depth predictor onto the adapted domain maps. Thus, due to the differentiability of the corresponding loss function, the training can be efficiently implemented using standard optimization tools. However, such indirect training of map transforms and depth network is susceptible to overfitting. The loss function focuses on a minimal number of mixture functions in the proximity of ground truth, and it can significantly distort the structural correspondence among the original mixture functions. To alleviate this, we also regularize the loss function by penalizing the distance of the original and adapted map using KL (Kullback Leibler) divergence. Thus, the loss function for the joint training of map transforms, and depth layer is given as:

$$\mathcal{L}(\theta_{\mathrm{M}},\theta_{\mathrm{D}}) = -\sum_{i} \log \mathcal{M}_{A,\theta_{\mathrm{M}}}(\mathcal{D}_{\mathcal{T}_I^i,\mathcal{T}_L^i,\theta_{\mathrm{D}}}) + \lambda D_{\mathrm{KL}}(\mathcal{M},\mathcal{M}_A) \tag{1}$$

Here, θ_{M} are the parameters for map transformation, and θ_{D} are the parameters of the last layer of depth predictor. For a trajectory $\mathcal{T}$, $\mathcal{T}_I$ represents the set of input images and $\mathcal{T}_L$ corresponding pose labels. $\mathcal{M}$ is the original domain map, and $\mathcal{M}_A$ is the adapted map to compensate for inaccuracies of the lightweight predictor. Both $\mathcal{M}$ and $\mathcal{M}_A$ are represented as GMMs. $\mathcal{D}_{\mathcal{T}_I^i,\mathcal{T}_L^i,\theta_{\mathrm{D}}}$ is the projection of predicted depth map of trajectory image $\mathcal{T}_I^i$ to 3D space by pin-hole camera model and assuming camera pose at the ground truth label $\mathcal{T}_L^i$.

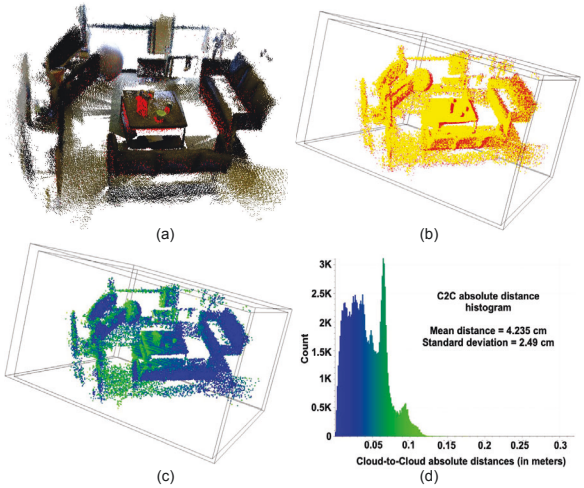

Fig. 3: (a) Merged original and transformed point-cloud maps. (b) Bounding boxes of the two maps with the original map in yellow and in red is the transformed map. (c) Relative distance coloring on the reference map data. (d) Histogram of cloud-to-cloud distances.

In (1), the regularization term requires computing the KL divergence between the original and adapted maps, namely $\mathcal{M}$ and $\mathcal{M}_A$ respectively. KL divergence of two Gaussian functions is defined in closed form but cannot be analytically extracted for two GMMs. In the proposed framework, original and adapted maps, $\mathcal{M}$ and $\mathcal{M}_A$, have the same number of mixture components, and with a strong enough regularization coefficient (λ), the relative correspondence among mixture functions maintains, i.e., for i$^{\text{th}}$ mixture function in $\mathcal{M}$, the nearest mixture function in $\mathcal{M}_A$ has the same index. Leveraging these attributes, the KL divergence of $\mathcal{M}$ and $\mathcal{M}_A$ can be approximated using Goldberger's approximation as [21]

$$D_{\mathrm{KL}}(\mathcal{M},\mathcal{M}_A) \approx \sum_i \pi_i \Big(D_{\mathrm{KL}}(M_i, M_{A,i}) + \log \frac{\pi_i}{\pi_{A,i}} \Big) \tag{2}$$

Here, M_i is the i$^{\text{th}}$ mixture component of $\mathcal{M}$, and $M_{A,i}$ is the corresponding component in $\mathcal{M}_A$. π_i is M_i's weight and $\pi_{A,i}$ is $M_{A,i}$'s weight. The KL divergence of M_i and $M_{A,i}$, i.e., $D_{\mathrm{KL}}(M_i, M_{A,i})$ is analytically defined. Thus, $D_{\mathrm{KL}}(\mathcal{M},\mathcal{M}_A)$ can be efficiently computed and is differentiable.

Fig. 3 shows the point cloud adaptations of Scene-02 in RGBD dataset [22] using the method. Fig. 3(a) contains both the original and adapted point-cloud (PC) maps. In Fig. 3(b), the reference or original map's 3D points are in yellow while the adapted PC is in red to highlight the adaptation difference. In Fig. 3(c), the reference point cloud's 3D points are color-coded based on the relative distance of corresponding points in the adapted map. The cloud-to-cloud (C2C) distance histogram is shown in Fig. 3(d). Thus, the results demon-

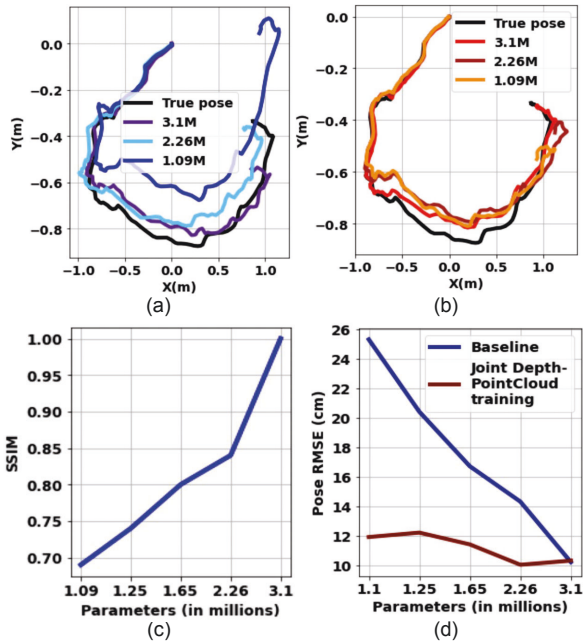

Fig. 4: (a) Predicted pose trajectory using the proposed integrated depth estimator and pose filter for varying depth network sizes without joint training of depth estimator and pose filter. (b) Pose predictions for varying depth network sizes using the proposed technique with jointly training learnable map transforms and lightweight depth predictor. (c) The structural similarity index of depth predictions reduces for a reduction in network size. (d) Comparison of Pose errors (RMSEs) for baseline and proposed technique.

strate that only a minimal tweaking of map data is sufficient to improve pose accuracy (evident in later results) despite extremely lightweight depth prediction.

3 Results and Discussion

Figs. 4(a) and (b) compare the predicted pose trajectory (for varying depth network size) from the proposed monocular localization against an equivalent framework where joint training of depth network and filtering model is not performed. The comparison uses the RGBD scenes dataset [22]. Fig. 4(c) shows the corresponding degradation in depth images, measured using structural similarity index measure (SSIM). In Fig. 4(d), despite significant degradation in depth image quality and reduction of depth predictor to one-third parameters, the proposed joint training maintains pose prediction accuracy by learning and adapting against systematic inaccuracies in depth prediction. Another crucial feature is that the original depth predictor can be trained on any dataset, and then tuned (on the last layer) for the application domain. For example, in the presented results, the original depth network was trained on NYU-Depth [23] and applied on RGBD scenes [22]. Thus, the predictor has access to vast training data that can be independent of application domain.

Fig. 5 demonstrates the resilience of proposed crossmodal pose prediction against DNN by considering extreme lighting variations. An equivalent MobileNet-v2-based PoseNet [1] is

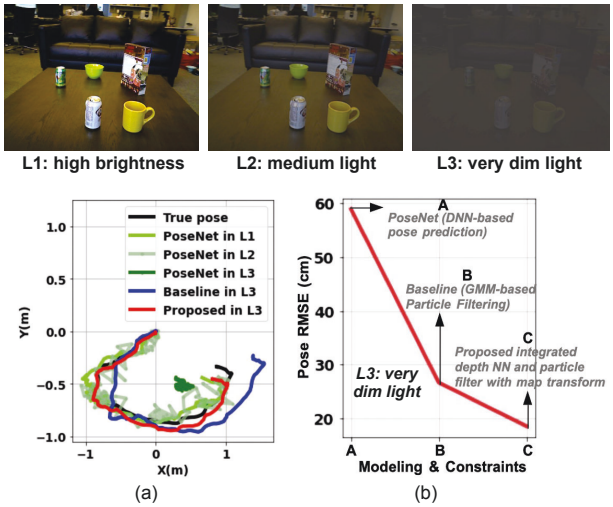

Fig. 5: On top: Indoor RGB image captured in different lighting conditions. (a) Comparison of pose trajectories for MobileNetv2-based PoseNet, baseline GMM map-based pose filtering, and proposed integrated framework of depth estimator and pose filter in various lighting conditions. (c) Pose error (RMSE) plot in very dim light for various models.

Table I: Comparison of pose trajectories from various scaled models

Localization model	# Parameters	Pose RMSE	Light variation tolerance
PoseNet [1]	1.7M	58.9 cm	Low
Traditional GMM-based	-	26.7 cm	Average
Proposed (no PC adaptation)	1.65M	16.7 cm	Good
Proposed (with PC adaptation)	1.8M	11.4 cm	Very good

utilized as DNN for the comparisons. On the top, input images are subjected to extreme lighting variations using models in [22] (L1: high brightness, L2: medium light, and L3: very dim light). Fig. 5(a) compares trajectories from PoseNet and our framework (with and without the joint training). In all cases, equivalent sized models are considered, shown in Table I. In Fig. 5(b), our framework is significantly more accurate than PoseNet in very dim light (L3) conditions due to in-built filtering loops, demonstrating superiority of crossmodal estimates than DNN-only estimates.

4 Conclusions

We presented a novel monocular localization framework by jointly learning depth estimates and map transforms. Compared to standard DNNs for pose estimates, the proposed approach is significantly more tolerant to model size scalability and environmental variations. Open representation of map and depth estimates in our approach also allows faster updates and resource efficiency by availing intermediate feature maps for other automation objectives.

Acknowledgement: This work was supported in part by COGNISENSE, one of seven centers in JUMP 2.0, a Semiconductor Research Corporation (SRC) program sponsored by DARPA.

5 References

[1] A. Kendall, M. Grimes, and R. Cipolla, "Posenet: A convolutional network for real-time 6-dof camera relocalization," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2938–2946.

[2] L. Zhou, Z. Luo, T. Shen, J. Zhang, M. Zhen, Y. Yao, T. Fang, and L. Quan, "Kfnet: Learning temporal camera relocalization using kalman filtering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 4919–4928.

[3] B. Wang, C. Chen, C. X. Lu, P. Zhao, N. Trigoni, and A. Markham, "Atloc: Attention guided camera localization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 06, 2020, pp. 10 393–10 401.

[4] D. Fox, S. Thrun, W. Burgard, and F. Dellaert, "Particle filters for mobile robot localization," in *Sequential Monte Carlo methods in practice*. Springer, 2001, pp. 401–428.

[5] P. Skrzypczyński, "Mobile robot localization: Where we are and what are the challenges?" in *International Conference Automation*. Springer, 2017, pp. 249–267.

[6] S. Thrun, "Probabilistic robotics," *Communications of the ACM*, vol. 45, no. 3, pp. 52–57, 2002.

[7] A. N. Dhawale, "Hierarchical gaussian distributions for real-time slam," Master's thesis, Carnegie Mellon University, Pittsburgh, PA, May 2020.

[8] S. Thrun, "Particle filters in robotics." in *UAI*, vol. 2. Citeseer, 2002, pp. 511–518.

[9] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 270–279.

[10] D. Wofk, F. Ma, T.-J. Yang, S. Karaman, and V. Sze, "Fastdepth: Fast monocular depth estimation on embedded systems," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019.

[11] N. Zioulis, A. Karakottas, D. Zarpalas, and P. Daras, "Omnidepth: Dense depth estimation for indoors spherical panoramas," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 448–465.

[12] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 3828–3838.

[13] V. Guizilini, J. Li, R. Ambrus, S. Pillai, and A. Gaidon, "Robust semi-supervised monocular depth estimation with reprojected distances," in *Conference on robot learning*. PMLR, 2020, pp. 503–512.

[14] D. Floreano and R. J. Wood, "Science, technology and the future of small autonomous drones," *nature*, vol. 521, no. 7553, pp. 460–466, 2015.

[15] V. Nekrasov, T. Dharmasiri, A. Spek, T. Drummond, C. Shen, and I. Reid, "Real-time joint semantic segmentation and depth estimation using asymmetric annotations," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 7101–7107.

[16] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[17] A. Dhawale, K. S. Shankar, and N. Michael, "Fast monte-carlo localization on aerial vehicles using approximate continuous belief representations," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5851–5859.

[18] H. Moravec and A. Elfes, "High resolution maps from wide angle sonar," in *Proceedings. 1985 IEEE international conference on robotics and automation*, vol. 2. IEEE, 1985, pp. 116–121.

[19] T. Schops, T. Sattler, and M. Pollefeys, "Bad slam: Bundle adjusted direct rgb-d slam," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 134–144.

[20] M. Heydari and Z. Duan, "Don't look back: An online beat tracking method using rnn and enhanced particle filtering," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 236–240.

[21] J. R. Hershey and P. A. Olsen, "Approximating the kullback leibler divergence between gaussian mixture models," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, vol. 4. IEEE, 2007, pp. IV–317.

[22] K. Lai, L. Bo, and D. Fox, "Unsupervised feature learning for 3d scene labeling," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 3050–3057.

[23] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *European conference on computer vision*. Springer, 2012, pp. 746–760.