A Bias-Aware Learning Model for Reliable Monocular Perception under Camera Misalignment

Richard Wang, Grant Han, Alexander Liu, Zixuan Ye, Alexander Deng, Lexie Liu and Xing Wei*

VEX V5RC Team 1698V,
BASIS Independent Silicon Valley Upper School, San Jose,
California, US (richardbojinwang2008, 26ghan,
alexanderliu089, zixuanye.sh, alexanderdeng11, lexieliu689,
xingweihome@gmail.com)
* Corresponding author

Abstract: Monocular vision systems are widely used in robotics and autonomous platforms for tasks such as distance estimation, object detection, and semantic understanding. However, these systems are highly sensitive to camera perturbations, such as physical misalignments caused by collisions, vibrations, or mechanical shocks during operation. Such perturbations introduce structural biases into the vision input, which significantly degrade model performance. In this work, we propose a self-adaptive, bias-aware deep learning framework designed to enhance the robustness of vision-based models under camera misalignment conditions. Our approach introduces a lightweight bias modeling module that learns to capture and compensate for systematic deviations caused by camera displacement. While our framework is applicable to various vision tasks, we evaluate it on monocular distance estimation to demonstrate its effectiveness. We construct a hybrid dataset simulating multiple camera perturbation scenarios and quantitatively assess the framework's robustness and generalization. Experimental results show substantial improvements over baseline models in terms of accuracy and stability under misaligned conditions. This work contributes a generalizable method to improve the resilience of vision-based systems operating in dynamic and unpredictable environments.

Keywords: Robust Vision, Sensors and Signal Processing.

1. INTRODUCTION

Accurate visual perception is essential for robotic and autonomous systems operating in unstructured real-world environments. Among visual tasks, monocular distance estimation is particularly appealing due to its low hardware complexity and cost-efficiency [1, 2]. However, in practical deployment, camera systems are often exposed to physical disturbances—such as impacts, vibrations, or structural shifts—that lead to unintended changes in camera pose [3]. These changes, including translations, rotations, and their combinations, introduce systematic distortions into the visual input. As a result, the performance of perception models, especially those relying on fixed geometric priors, degrades significantly. This challenge is well-documented in the robotics field, as noted by [14], who showed how mechanical impacts and pose drift can lead to perceptual failure in mobile robot navigation.

Most existing monocular vision models are trained under fixed camera assumptions and do not incorporate mechanisms to detect or compensate for deviations from the original configuration. This limits their robustness in dynamic or unpredictable environments, as highlighted in [15], which analyzed how structural changes in the visual scene can disrupt learned features and localization pipelines.

To address this challenge, we propose a bias-aware, self-adaptive framework for monocular distance estimation. The core idea is a two-tower neural architecture: a Nominal Tower handles the primary task under calibrated conditions, while a lightweight Bias Modeling Tower learns to correct for deviations introduced by camera misalignment. During training, these towers are optimized in two stages—first under nominal conditions, then with synthetic or real misalignment data—enabling modular design and efficient deployment.

At runtime, the system performs self-adaptive inference by detecting misalignment and activating the appropriate bias correction pathway. This can be done via exhaustive matching against known misalignment cases or via a predictive model trained to classify visual disturbances. The architecture ensures high accuracy under normal conditions and robust correction when needed, with minimal computational overhead.

We evaluate our method using a real-world robotics platform (VEX V5 [5]), simulating a variety of camera displacement scenarios. Experiments show that our bias-aware model improves distance estimation accuracy under misaligned conditions. The proposed method represents a practical step toward self-healing, configuration-robust monocular vision systems suitable for use in unstable or reconfigurable robotic platforms.

The remainder of this paper is organized as follows. Section II reviews related work in monocular distance estimation and robust vision modeling. Section III presents the proposed bias-aware learning framework and its two operational modes. Section IV describes the experimental setup, comparison strategy, and evaluation protocols. Section V reports and discusses the experimental results. Finally, Section VI concludes the paper and suggests directions for future research.

2. RELATED WORK

2.1 Monocular Distance Estimation

Monocular depth estimation has been an active area of research due to its practical advantages in cost and hardware simplicity. Early supervised models like [6] proposed multi-scale convolutional networks trained with RGB-depth datasets, achieving high accuracy but requiring dense ground-truth depth maps. To overcome data limitations, self-supervised methods emerged that exploit photometric consistency between consecutive frames [7] or stereo pairs [8], removing the need for direct supervision.

Despite impressive progress, these methods assume a stable and calibrated camera pose, which is not always guaranteed in real-world applications involving physical impacts or mechanical drift. This assumption makes the models brittle to perturbations that shift the spatial distribution of image content.

2.2 Robustness to Perturbations

Improving model robustness under varying environmental conditions has been a central concern in both computer vision and robotics. One common strategy is domain randomization, where training data is artificially augmented with randomized lighting, textures, or viewpoints to increase generalization [9]. Similarly, viewpoint augmentation strategies [10] introduce pose diversity to reduce sensitivity to camera angle variations.

More recent methods incorporate geometric priors to encode structural information in the scene. For example, StruMonoNet [16] uses surface normals and depth gradients to constrain monocular predictions, improving consistency across spatial perturbations. However, these models typically assume small, statistically random deviations, and lack mechanisms to systematically interpret or correct deterministic camera shifts.

2.3 Calibration and Misalignment Compensation

Explicit modeling of camera miscalibration is less explored in monocular setups. CalibNet [11] introduced a deep learning approach for extrinsic camera-LiDAR calibration between a 2D camera and a LiDAR sensor; FusionDepth [12] and related works integrate data from LiDAR, radar, or stereo to correct monocular errors, but the techniques applied limit their applicability in lightweight and/or self-adaptive systems.

Table 1 summarizes the key differences between CalibNet and our approach. While both ultimately improve distance estimation, they do so in distinct ways: CalibNet focuses on multi-sensor geometric alignment, whereas our bias-aware two-tower framework explicitly compensates for monocular misalignment at inference time. This distinction makes our method particularly suitable for cost-sensitive and adaptive deployments where auxiliary sensors are unavailable at runtime.

Recent robotics-focused approaches like RoVi-Aug [13] take a complementary path, improving robustness by augmenting training data with simulated viewpoint shifts. While effective for enhancing generalization, these approaches do not explicitly model or correct systematic camera bias at inference time. Our framework and RoVi-Aug are therefore complementary: augmentation can broaden the training distribution, while our bias-aware correction ensures runtime adaptation to structured misalignment.

Table 1.Comparison between CalibNet and the bias-aware framework.

Aspect	Bias-Aware Two Tower	CalibNet
Goal	Online correction for camera misalignment	Extrinsic calibration between camera and LiDAR for depth accuracy
Inference Setup	Single RGB camera	2D Camera + LiDAR
Cost	Lightweight	Extra hardware and intensive computation
Accuracy	Restores nominal accuracy	High geometric accuracy
Generaliz ation	Applicable beyond distance estimation (e.g, detection, segmentation, etc.)	Narrowly designed for distance estimation

3. METHOD

3.1 Problem Overview

Monocular vision systems are widely adopted in robotics and autonomous navigation due to their cost-effectiveness and simplicity. However, these systems are highly sensitive to changes in camera pose caused by physical disturbances, such as collisions or mechanical impacts during operation. These misalignments introduce systematic biases into the visual input, significantly degrading the performance of downstream perception tasks, such as distance estimation.

We define the problem as learning to compensate for such structural biases in a way that is robust, efficient, self-adaptive, and suitable for real-time deployment. Beyond handling unexpected misalignments, our approach is also extensible to controlled, expected changes in camera configuration (e.g., modular robots with reconfigurable sensors). The goal is to make the model resilient to these deviations with minimal retraining or inference overhead.

3.2 Bias-Aware Learning Framework

To address this, we propose a bias-aware learning framework based on a two-tower neural network architecture, as illustrated in Fig 1. The design separates the processing of nominal and misaligned camera conditions into two distinct yet cooperative towers:

Nominal Tower: This tower handles inputs from the camera in its correct, calibrated position. It is responsible for learning the primary vision task — in our experiments, monocular distance estimation. While our experimental setup uses a 3-layer convolutional neural network (CNN) to focus on evaluating the bias correction mechanism, the Nominal Tower is not inherently constrained in complexity. In practice, it can adopt any state-of-the-art visual backbone (e.g., ResNet, Vision Transformer) depending on the target application's accuracy and latency requirements. This design choice ensures that the proposed framework remains broadly compatible with existing vision models, and that improvements in base model performance can be naturally integrated without altering the bias correction structure.

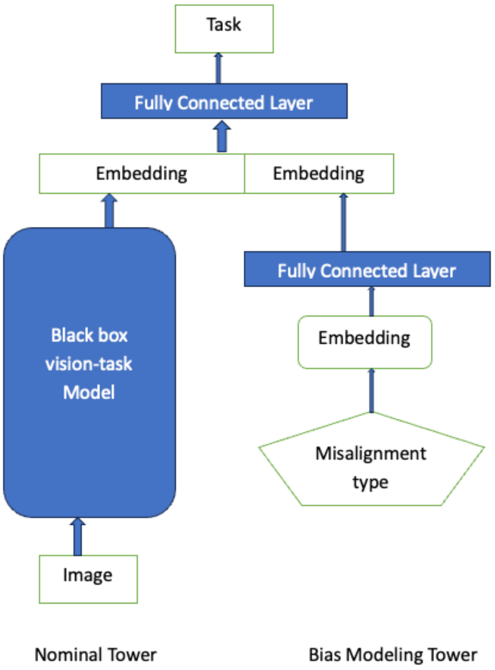

Fig. 1. Bias-aware learning framework.

Bias Modeling Tower: The second tower is a lightweight module that models the effect of camera misalignment on the vision task. Its input is from a misaligned camera position, and its role is to capture the systematic transformation between the misaligned input and the output expected from the nominal tower. This tower is intentionally kept simple to ensure low latency and fast adaptation for online deployment.

The outputs of the two towers are passed through a shared linear fusion layer, which combines their embeddings and produces a final task-specific output (e.g., estimated distance). This fusion acts as a bias-correction mechanism.

3.3 Training Strategy

The training process is designed to balance optimal performance under nominal camera conditions with robustness against unexpected misalignment. It is divided into two distinct phases:

In the first phase, the model is trained using data collected with the camera in its correct, calibrated position. During this stage, the bias modeling tower is masked out, reducing the architecture to a standard single-tower model. This allows the system to be optimized solely for nominal conditions, achieving the best possible performance and inference speed without any overhead from the bias correction module. Since most real-world operations occur under normal conditions, this ensures efficiency and reliability in the majority of cases.

In the second phase, the bias modeling tower is activated and trained, while the original vision tower from the first phase is frozen. This design ensures that the core model remains unchanged and can continue to serve as the fast-path inference engine. The bias tower is trained using data collected under representative camera misalignment conditions, where it learns to estimate and compensate for the systematic deviations caused by displacement.

Because collecting real-world misalignment data is expensive and time-consuming, we use a smaller dataset in this phase. Despite this limitation, the bias-aware module demonstrates strong generalization across different types of displacement, due to the structural separation of task modeling and bias estimation.

This modular training strategy enables:

- Fast and accurate inference under normal conditions, by defaulting to the original single-tower model.
- Flexible bias compensation when misalignment is detected, with minimal additional latency.
- Ease of deployment and maintenance, as the two components can be trained and updated independently. The numbering of the subsection should take the above form.

3.4 Self-Adaptive Inference Mechanism

To support real-world deployment in dynamic and potentially unstable environments, the proposed framework includes a self-adaptive inference mechanism that dynamically adjusts to camera misalignment at runtime.

Under normal conditions—when the camera remains in its calibrated position—the system operates in a lightweight single-tower mode, using only the original task model. This ensures low-latency and high-efficiency inference suitable for time-critical robotics applications. However, when the system detects or suspects a misalignment (e.g., due to collision, internal odometry changes, external feedback, or degraded prediction confidence, etc.), it automatically activates the bias-aware correction process. Our framework performs a two-step self-adaptation process:

1. Misalignment Identification: The system estimates the new camera pose or its effect on output through one of two methods:

- Method A (Exhaustive Matching): The system compares the current visual input against embeddings generated from all trained misalignment cases and selects the one that minimizes the discrepancy between the nominal(pre-misalignment) and observed(post-misalignment) outputs. This method is feasible because the Bias Modeling tower is lightweight, allowing efficient comparisons across cases. Additionally, the Nominal tower—which performs the more computationally intensive operations—needs to be computed only once per input. The computation is detailed in Equation (1).

$$m^* = \arg\min_{m_i \in \mathcal{M}} \left| \overline{f_{\text{biased}}}(x_{\text{biased}}, m_i) - f_{\text{orig}}(x_{\text{orig}}) \right| \quad (1)$$

Here:

m^*: the best-matched misalignment configuration

$\mathcal{M}$: the set of possible misalignments

x_{orig}: the image input before camera misalignment

x_{biased}: the current input from the misaligned camera

f_{orig}: the computation by the nominal tower only

f_{biased}: the computation by the two tower model with bias correction.

To achieve optimal performance, we compute the mean of f_{biased} on a few vision samples collected after the misalignment, comparing them with the original distances measured before misalignment. This approach accepts the risk of distance variation caused by the robot's continued motion. We analyze this trade-off in the Experiments section. Equation (2) presents the computation.

$$\overline{f_{\text{biased}}}(x_{\text{biased}}, m_i) = \frac{1}{k} \sum_{t=t_0}^{t_0+k-1} f_{\text{biased}}^{(t)}\left(x_{\text{biased}}^{(t)}, m_i\right) \quad (2)$$

t_0 is the current timestep and k is the size of samples used for computation.

- Method B (Predictive Mapping): An auxiliary machine learning model can be trained to directly predict the misalignment type from visual cues observed before and after the disturbance. Compared to Method A, this approach is potentially more scalable, but it requires additional training data and model complexity. Our proposed modeling setup is as follows:

Training features: Two images (before and after the disturbance)

Training label: Misalignment type

Model setup: Multi-class classification

Model structure: Two-tower CNN (one for each image) with a multi-layer DNN applied after concatenating the embeddings from the two towers

Loss function: Softmax cross-entropy

Model Design. 1. Predictive mapping model.

Since our experiments primarily focused on the two-tower regression framework in Method A, we chose it for simplicity in this study. However, we regard Method B as a promising direction for future exploration.

2. Bias-Corrected Inference: Once the misalignment effect is identified, the system activates the corresponding pathway in the bias modeling tower to produce a corrected output. Because the bias modeling tower is lightweight, this correction adds negligible latency and is suitable for real-time applications.

4. EXPERIMENTS

4.1 Experimental Setup

To evaluate the effectiveness and adaptability of our proposed bias-aware learning framework, we implemented and tested it on a mobile robotic platform using the VEX V5[5] system. The robot is equipped with a forward-facing monocular camera for visual distance estimation. Ground truth distances were obtained using the robot's odometry system, which provided accurate labels for each frame under various configurations.

To simulate realistic camera misalignment scenarios, we applied controlled mechanical shocks to the robot and intentionally loosened the camera mount on the VEX V5 platform. From the resulting variations, we manually selected six representative types of camera displacement, including isolated translations, isolated rotations, and compound combinations of both translation and rotation. These cases were chosen to provide a diverse yet manageable set of perturbations for evaluating the proposed framework.

These perturbations reflect common conditions caused by collisions or vibrations in real-world robotic applications. While more combinations are possible in practice, we selected these six types for experimental manageability and simplicity.

For each misalignment type, we collected a dedicated dataset in our indoor test environment. The misaligned camera input was paired with accurate distance labels from the odometry system and used both for training and evaluation.

4.2 Baseline and Comparison Strategy

To establish a performance baseline, we first collected training and testing data under the original, unaltered camera alignment condition (denoted as Orig). A standard deep regression model was trained on this dataset without the bias-aware module. This setup represents a typical deployment scenario in which the model is trained in clean, controlled conditions. Importantly, this version of the model does not incur any latency or computational overhead from the bias-aware module, making it directly comparable in runtime efficiency. In practical applications, such a model is typically trained on a large-scale dataset, substantially larger than the datasets available for misalignment conditions.

We then evaluated this Orig model on datasets collected under all six misaligned conditions to measure the impact of misalignment on inference performance. Finally, we applied our bias-aware framework to the same perturbed inputs and compared the results. Our goal is to demonstrate that the proposed method can significantly recover performance loss induced by misalignment, even when the original model fails.

For comparison, we also include an Offset Calibration baseline, which applies a simple additive correction to the model outputs as a post-processing step. All models share the same base architecture to ensure a fair evaluation.

4.3 Evaluation Metrics

We evaluate performance in the following aspects:

- Accuracy of misalignment matching
- Mean Absolute Error (MAE) between predicted and ground-truth values.
- Relative Diff (%)

5. RESULTS AND ANALYSIS

We evaluate our proposed framework in two stages, as outlined in Section 3.4: (1) Misalignment Identification and (2) Bias-Corrected Inference. The experiments demonstrate both the sensitivity of monocular systems to camera misalignment and the effectiveness of our bias-aware learning framework in restoring degraded performance.

5.1 Misalignment Identification

Case A: Vision input only. To test the system's ability to infer the current misalignment condition based solely on visual input, we performed exhaustive matching using 1, 2, and 3 post-misalignment vision samples (as per Equation 2). Results are shown in Table 2.

Table 2.Misalignment identification accuracy using vision input only.

	1 sample accuracy	2 sample accuracy	3 sample accuracy
M1	31.9%	32.9%	33.4%
M2	43.2%	42.1%	41.9%
M3	77.1%	77.8%	77.7%
M4	50.0%	50.4%	50.0%
M5	62.8%	62.7%	62.6%
M6	39.2%	42.1%	41.7%
AVG on all M sets	50.7%	59.8%	59.8%

Performance varies significantly across misalignment types. M1, which involves only a small translation, shows the lowest identification accuracy due to its minimal deviation from the original alignment, making it difficult to distinguish. M6, despite involving a rotation, also performs poorly—likely for a similar reason, as the visual shift remains subtle and hard to differentiate from the unperturbed condition.

Case B: Additional sensor. To assess the potential of multi-sensor input, we repeated the identification experiment using ground-truth distance from the odometry system in place of f_{orig} in Equation (1). We got accuracy as in Table 3. The accuracy has increased significantly for all segments. Table 3 shows the significant accuracy improvements achieved with this sensor fusion strategy.

Table 3.Misalignment identification accuracy by monovision + other sensor.

	1 sample accuracy	2 sample accuracy	3 sample accuracy
M1	32.6%	43.8%	40.5%
M2	44.2%	73.1%	87.3%
M3	77.4%	82.2%	89.0%
M4	49.7%	71.6%	82.2%
M5	63.2%	80.3%	90.8%
M6	39.5%	69.1%	82.2%
AVG on all M sets	51.1%	70.0%	78.7%

These results demonstrate that fusing visual input with a secondary sensor dramatically improves identification performance—especially when using multiple samples.

Compare Case A and B: Figure 2 visualizes the improvement in accuracy across all misalignment types by comparing Table 2 and Table 3.

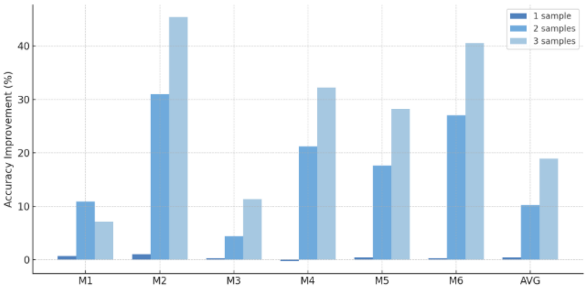

Fig. 2. Accuracy improvement from additional sensor (odometry) across misalignment types.

We believe further gains are possible by incorporating a confidence score into the model's predictions. This would allow the matching algorithm to selectively trust only highly confident predictions.

Figure 3 is a comparative bar graph showing how the number of vision samples impacts misalignment identification accuracy under two conditions: vision-only and vision with an additional sensor.

The identification accuracy improves notably as the number of samples increases, particularly when external sensor data is incorporated. From the results in Table 2 (vision input only), increasing the sample size from 1 to 2 yields moderate gains, with average accuracy rising from 50.7% to 59.8%. However, the improvement saturates beyond two samples, suggesting diminishing returns in vision-only scenarios. This is likely due to the vision-only system relying on a single distance measurement taken just before misalignment occurs, which becomes increasingly unreliable as more samples are collected—since the robot continues to move during data acquisition. In contrast, the inclusion of external sensor data (Table 3) leads to substantial accuracy gains across all misalignment cases. With sensor fusion, the average accuracy increases from 51.1% with one sample to 78.7% with three samples, demonstrating the value of complementary information beyond monocular vision.

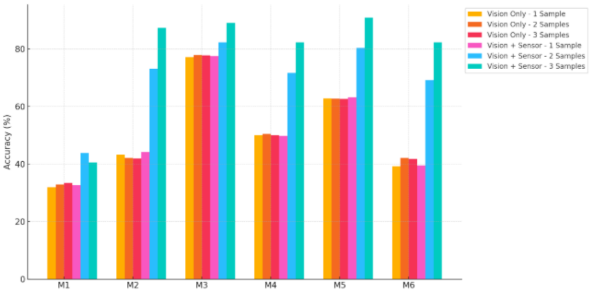

Fig. 3. Effect of sample count on misalignment identification accuracy

The impact of sample count varies across misalignment types. For example, M3 and M5 benefit significantly from both additional samples and sensor integration, achieving over 89% accuracy with three samples and sensor input. In contrast, M1 shows minimal improvement due to its subtle nature, and M6, despite being a compound misalignment, also sees only modest gains. This suggests that certain perturbations remain challenging to differentiate based on visual input alone and benefit less from temporal aggregation.

Overall, the results demonstrate that both multi-frame evidence and sensor fusion can meaningfully enhance robustness in misalignment detection, though their benefits depend on the nature of the perturbation.

5.2 Bias-Corrected Inference

To evaluate the effectiveness of our bias-aware inference module, we compared the predicted distances from different models to the ground-truth distances measured by the robot's odometry system. The results are reported in Table 4 using the Mean Absolute Error (MAE) metric.

To isolate the effect of bias correction and avoid confounding results due to misclassification errors, we used the true misalignment labels for each input during this evaluation. That is, the bias-aware module was provided with the correct misalignment type rather than relying on the identification step described earlier. This ensures that the performance gain reported here reflects only the inference module's correction capability.

We additionally included a comparison with an Offset Calibration baseline in Table 5, which applies a simple affine transformation (i.e., a post-hoc additive correction) to the model's outputs. This serves as a lightweight correction strategy for systematic prediction errors.

Table 4. Bias correction performance with proposed model.

	MAE (Orig model only)	MAE (Bias-aware model)	Error diff
Orig	2.62	X	X
M1	31.80	14.04	-55.8%
M2	55.38	8.91	-83.9%
M3	82.20	22.89	-72.1%
M4	106.37	14.41	-86.5%
M5	97.79	8.56	-91.2%
M6	21.80	6.75	-69.0%
All M sets	62.34	12.78	-79.5%

These results demonstrate that the bias-aware inference model significantly outperforms the offset calibration baseline. While offset calibration yields only marginal or inconsistent improvements—and even increases error in some cases—our proposed method consistently and substantially reduces prediction error across all misalignment settings. On average, the bias-aware model achieves a 79.5% reduction in MAE, confirming its robustness and effectiveness even under severe camera misalignment scenarios (e.g., M4 and M5). This supports our hypothesis that misalignments induce structured distortions in feature space that can be systematically corrected by explicitly modeling and compensating for bias conditions during inference.

Table 5. Bias correction performance with offset calibration.

	MAE (Orig model only)	MAE (Bias-aware model)	Error diff
Orig	2.62	X	X
M1	31.80	34.82	+9.5%
M2	55.38	54.82	-1.0%
M3	82.20	83.15	+1.2%
M4	106.37	108.09	+1.6%
M5	97.79	92.35	-5.6%
M6	21.80	21.72	-0.4%
All M sets	62.34	65.16	+4.5%

Overall, bias-aware inference proves highly effective at recovering performance lost due to camera misalignment. Across all misaligned conditions, MAE was reduced by over 80% on average compared to the original model trained solely under ideal conditions, highlighting the significant advantage of addressing bias directly during inference.

Notably, offset calibration improvements are inconsistent, sometimes worsening performance, even for extreme misalignment cases such as M5 and M6. In contrast, the bias-aware approach consistently maintains robustness and adaptability, validating the core assumption that camera misalignment introduces systematic biases in feature distributions which can be effectively compensated by explicitly modeling the perturbation space.

6. CONCLUSION

In this work, we introduced a general bias-aware learning framework designed to improve the robustness of vision-based models under sensor misalignment and structured input perturbations. While our experimental validation focused on a visual perception task using monocular camera input on a mobile robotic platform, the approach is broadly applicable to many computer vision problems where feature distribution shifts degrade model performance.

Our framework operates in two stages: (1) identifying the underlying bias condition (e.g., caused by physical misalignment), and (2) applying a learned correction during inference conditioned on the identified bias. Experimental results demonstrate that this approach significantly improves model reliability under real-world perturbations. By integrating additional sensor information and leveraging multi-frame evidence, our method achieves high accuracy in identifying misalignment conditions. The bias-aware inference module then effectively compensates for the resulting feature shifts, yielding substantial performance gains compared to traditional models and simple calibration baselines.

These findings demonstrate that structured biases—such as those caused by sensor displacement—can be systematically modeled and corrected, leading to more adaptive and reliable deployment of vision systems in dynamic environments. This has broad implications for applications in robotics, autonomous navigation, inspection, and other domains where consistent sensor alignment cannot be guaranteed.

Future work will focus on further improving the accuracy of the framework, extending its application to more diverse tasks such as object detection, segmentation, and 3D reconstruction, and exploring self-supervised strategies for online bias modeling without requiring manual labeling.

REFERENCES

[1] D. Eigen, C. Puhrsch, and R. Fergus, “Depth Map Prediction from a Single Image using a Multi-Scale Deep Network”. NIPS, 2014

[2] C. Godard, O.M. Aodha, and G. J. Brostow, “Unsupervised Monocular Depth Estimation with Left-Right Consistency”. CVPR, 2017.

[3] S. Scherer, S. Singh, L. Chamberlain, and M. Elgersma, “Flying Fast and Low Among Obstacles: Methodology and Experiments”. The International Journal of Robotics Research, 2008.

[4] A. Krizhevsky, I. Sutskever, and G.E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks”. NIPS 2012.

[5] https://www.vexrobotics.com/v5?srsltid=AfmBOorPEo_UTSvhlsbsBafiqt-XMKN8IC892mRHz558b3glDFihc_T_

[6] D. Eigen, C. Puhrsch, and R. Fergus, “Depth map prediction from a single image using a multi-scale deep network,” NeurIPS, vol. 27, pp. 2366–2374, 2014.

[7] .R. C. Baker and B. Charlie, “Nonlinear unstable systems,” International Journal of Control, vol. 23, no. 4, pp. 123-145, 1989.

[8] R. Garg, B. Vijay Kumar, G. Carneiro, and I. Reid, “Unsupervised CNN for Single View Depth Estimation: Geometry to the Rescue,” in Proceedings of the European Conference on Computer Vision (ECCV), 2016, pp. 740–756.

[9] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, “Domain randomization for transferring deep neural networks from simulation to the real world.” IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 23–30, 2017.

[10] J. Tremblay, A. Prakash, D. Acuna, M. Brophy, V. Jampani, C. Anil, T. To, E. Cameracci, S. Boochoon, and S. Birchfield, “Training deep networks with synthetic data: Bridging the reality gap by domain randomization”. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 1082–1090, 2018.

[11] A. G. Loquercio, W. Luo, A. Geiger, and R. Siegwart, “CalibNet: Geometrically supervised extrinsic calibration using 3D spatial transformer networks,” IROS, pp. 1110–1117, 2019.

[12] A. Valada et al., “Self-supervised model adaptation for multimodal semantic segmentation”. International Journal of Computer Vision(IJCV), vol. 128, no. 5, pp. 1239–1285, 2019.

[13] Y. Yue et al., “RoVi-Aug: Robust vision-based learning through viewpoint-aware data augmentation,” IEEE Robotics and Automation Letters, vol. 7, no. 2, pp. 3603–3610, 2022.

[14] P. Fankhauser, M. Bloesch, D. Rodriguez, R. Kaestner, M. Hutter, and R. Siegwart, “Kinect v2 for Mobile Robot Navigation: Evaluation and Modeling,” Sensors, vol. 16, no. 9, 2016.

[15] T. Sattler, W. Maddern, C. Toft, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic, F. Kahl, and T. Pajdla, “Benchmarking 6DOF Outdoor Visual Localization in Changing Conditions,” IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 8601–8610.

[16] Z. Yang, L.E. Li, and Q. Huang, “StruMonoNet: Structure-Aware Monocular 3D Prediction”. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7413–7422, 2021.