Deep learning-based covariance estimation for relative pose measurements

Alireza Ahrabian and Quan Nguyen and Nikos Toulios and Anthony Ohazulike

***Abstract*— We propose a general covariance estimation method for relative pose measurements using deep learning. Our approach extends previous system specific covariance estimation models. Such models map input images acquired from two different viewpoints to a covariance estimate. While such models have successfully been applied to relative pose measurements obtained from visual odometry, the extension to the general system scenario is rather more challenging. In this paper, we propose to map both the inputs images acquired from two viewpoints along with the relative pose measurement to a covariance estimate. By including the relative pose measurement as an additional input to the mapping, we show that it is possible to predict covariance for general relative pose measurements.**

I. INTRODUCTION

Covariance estimation is a challenging task for non-stationary error distributions, this is particularly true for covariance estimation for relative pose measurements; that is, the motion of a robot between two different time instances. Fusion of relative pose measurements obtained from either a single system or from multiple systems is important for a variety of tasks related to robotic localisation and mapping [1][2][3]. The inclusion of uncertainty (covariance) in the measurement fusion process can reduce estimation errors via filtering or smoothing. However, estimating the uncertainty of relative pose measurements can be a challenging task due to the dependence of uncertainty on the system utilised to extract such a measurement. This is exemplified by relative pose measurements obtained from visual odometry (VO) where it is apparent that the error distribution can vary based on the texture [4], as well as the number of dynamic objects in the scene.

As a result, the work in [5] proposed a CNN based covariance estimator of relative pose measurements obtained via visual odometry. This was achieved by mapping the input images (as processed by VO) to a covariance estimate. Furthermore, the authors in [5] hypothesised that their method could be extended to the uncertainty estimation of an arbitrary relative pose measurement system (albeit this was not clearly explored in the paper). The work in [7] proposed an extension of [5] by estimating both a correcting relative pose and a covariance using a modified loss function along with larger CNN model. Whilst the estimation of a correcting pose is outside the scope of this paper, the work demonstrated good results on covariance estimation when applied to the KITTI dataset.

This research has been conducted as part of the EVENTS project, which is jointly funded by the European Union, under grant agreement No 101069614 and Hitachi Astemo Ltd. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Commission. Neither the European Union nor the granting authority can be held responsible for them.

The authors are members of Hitachi Europe Limited.

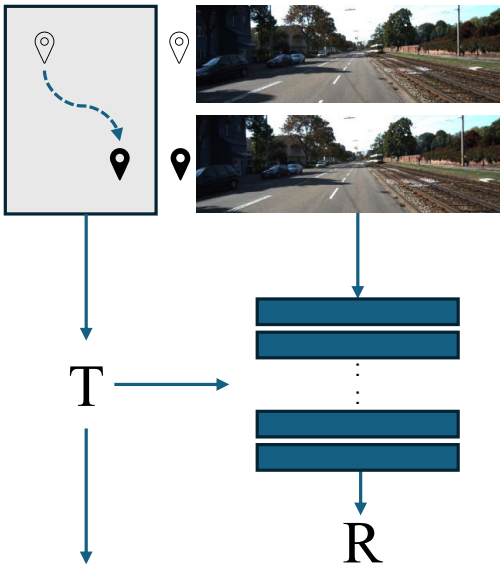


Fig. 1: Proposed covariance estimation method for relative pose measurements. The model processes both images acquired from two different views along with the measured relative pose between the two views.

In this work we propose a general covariance estimator for relative pose measurements obtained from an arbitrary system. We achieve this by extending the method in [7] as follows, 1) we propose to process both input images acquired from two viewpoints along with the measured relative pose (shown in Fig. 1), and 2) following the work in [11] we assume independent and identically distributed translation and orientation parameters (i.e. a diagonal covariance matrix) in order to reduce the number of redundant covariance parameters being inferred. Our proposed model is capable of predicting covariance of the relative pose measurements without the need to train specialised models for a specific sensing system. We demonstrate both quantitatively and qualitatively the performance of our proposed model on the KITTI dataset.

II. PRELIMINARIES

In this work we consider robotic motion with 6 degrees of freedom (3 translation and 3 rotation parameters), that is, motion in 3D space. Furthermore, for a given sensor s and time instants i and $i+1$ we have a relative pose

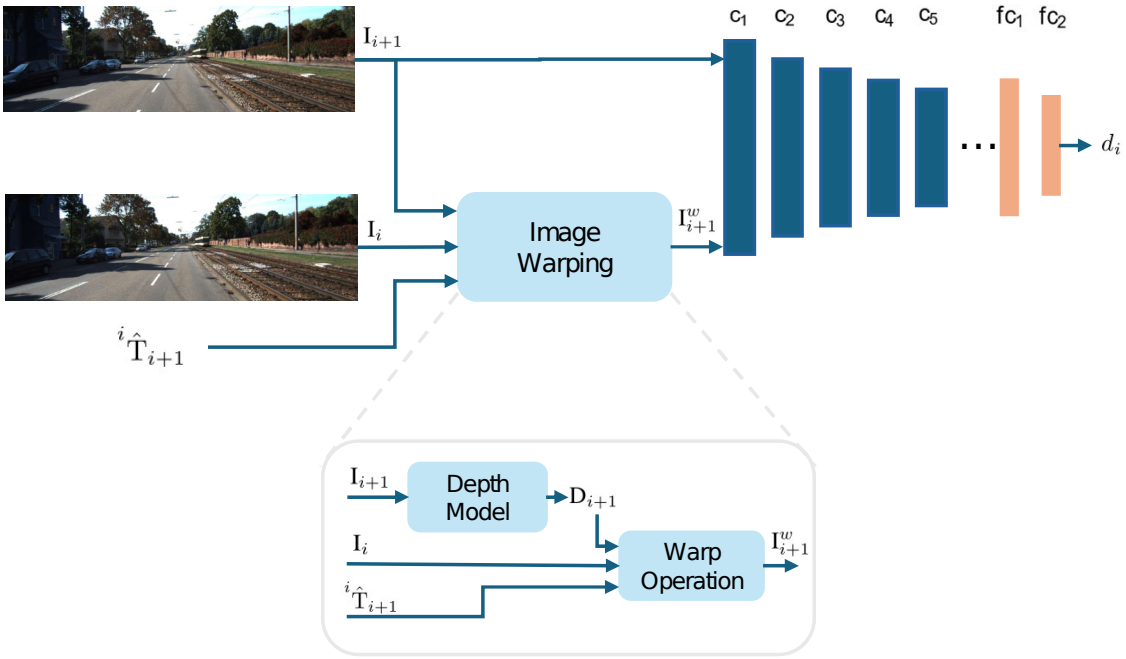


Fig. 2: Proposed covariance prediction model that computes covariance for a general relative pose measurement. The input consists of images acquired from two different viewpoints along with the measured relative pose, while the output is the estimated covariance. The model utilises the measured relative pose via an image warping operation, thereby enabling the covariance estimate to vary according to measured relative pose.

measurement (or estimate) ${}^i\hat{\mathrm{T}}_{i+1} \in \mathbb{R}^{4\times4}$. We consider the following problem, namely, given the error of the relative pose measurement e_i (computed for sensor s), we aim to estimate the covariance $R_i \in \mathbb{R}^{6\times6}$ of the error distribution which is assumed to be Normally distributed [5],

$$e_i \sim \mathcal{N}(0, R_i), \tag{1}$$

where it should be noted that this formulation assumes that the error is independently distributed but not identically distributed, thereby capturing the heteroscedastic nature of relative pose measurement errors. The error e_i, is computed as follows,

$$e_i = \log({}^i\hat{\mathrm{T}}_{i+1}\, {}^i\mathrm{T}^{-1}_{i+1}), \tag{2}$$

where ${}^i\mathrm{T}_{i+1}$ is the ground truth relative pose and log is the logarithmic map from a pose matrix ($SE(3)$ group) to a pose vector ($\mathbb{R}^6$ vector) [9].

III. Problem Statement and Related Work

The primary aim of this work is to estimate the covariance R_i of the error e_i for a relative pose measurement (or estimate) obtained between two different time instants, i and $i+1$, that is **independent** of the sensor s (a relative pose measurement obtained from an arbitrary sensor) [6]. More formally, given an input that contains both a set of images $\mathcal{I}_i$ and relative pose measurement ${}^i\hat{\mathrm{T}}_{i+1}$ obtained from an arbitrary sensor s, we aim to predict an output that is the covariance,

$$R_i = g(\mathcal{I}_i, {}^i\hat{\mathrm{T}}_{i+1}). \tag{3}$$

Our problem formulation builds on existing work [5][7] that developed covariance estimation models around a specific sensor, namely VO relative pose measurements. The supervised learning problem formulated for the previous work was formulated as follows

$$R_i = g(\mathcal{I}_i). \tag{4}$$

While the formulation in (4) is able to learn mappings between the set of input images and the covariance for VO, generalisation to an arbitrary relative pose measurement system is more challenging (owing to the reliance on on the input data $\mathcal{I}_i$).

IV. Proposed Method

We now describe the details of the covariance prediction model shown in (3), where the input consists of following,

- Set of monocular[1] colour images $\mathcal{I}_i = \{\mathrm{I}_i, \mathrm{I}_{i+1}\}$ obtained at time indices i and $i+1$.
- The measured relative pose ${}^i\hat{\mathrm{T}}_{i+1}$.

The output is the predicted covariance R_i for the measurement ${}^i\hat{\mathrm{T}}_{i+1}$. The architecture of our proposed method is shown in Fig. 2 and consists of two functional stages. The first stage leverages the measured relative pose via the warping operation of the image obtained at time instant i (more detail in Section IV.A) and the second stage predicts the covariance given the warped image along with the raw image obtained at time instant $i+1$. In the following

[1]It should be noted that our work can easily be extended to stereo images.

subsections we provide further explanation on the respective stages.

A. Input Image Warping

Our proposed first stage warping operation enables the utilisation of the measured relative pose when predicting covariance. The warping operation is based on the spatial transformer method in [10] and aims to construct an image of the *target* view (i.e. the warped image), given: the *source* view image, the dense depth estimate of the target view and relative pose between the source and target view [12][13]. We consider the image acquired at time index i as the source view image, while image acquired at time index $i+1$ as the target view image. The key idea of the warping operation is to first project the pixel coordinate (p_{i+1}) in the target view image to a pixel coordinate (p_i) in the source view image,

$$p_i = \mathrm{K}\,{}^{i}\hat{\mathrm{T}}_{i+1}\mathrm{D}_{i+1}\mathrm{K}^{-1}p_{i+1}, \tag{5}$$

where K is the camera intrinsic matrix and D_{i+1} is the estimated depth for each pixel in I_{i+1}. An interpolation method (bilinear) is then used to compute an intensity value for the projected pixel, given nearby source pixels [10], yielding the warped image I^{w}_{i+1}.

B. Covariance Prediction

The covariance prediction network takes concatenated images $(\mathrm{I}_{i+1}, \mathrm{I}^{w}_{i+1})$ as input and produces the predicted covariance R_i. We follow the network architecture described in [7] (shown in Table. I), that is, a first stage convolutional encoder followed by a second stage fully connected regression head. Our proposed warping operation enables the utilisation of the estimated relative pose during covariance inference, owing to the warped image I^{w}_{i+1} varying according to ${}^{i}\hat{\mathrm{T}}_{i+1}$; that is, the network learns a mapping that considers the variation of I^{w}_{i+1} with respect to the image I_{i+1}.

The intuition for our proposed method can be motivated by the following simple example. Consider a stationary robot that has acquired the same image at two different time instances ($\mathrm{I}_i = \mathrm{I}_{i+1} = \mathrm{I}$). The robots measured relative pose is given by, ${}^{i}\hat{\mathrm{T}}_{i+1} = \epsilon$, where ϵ is perturbing pose matrix. Given that the ground truth relative pose is equal to the identity matrix (i.e. no motion), then the covariance is a function of the perturbing pose ϵ. The methods in [5][7] would predict a covariance that is invariant to the perturbing pose ϵ as the input to the networks are the concatenated raw images $(\mathrm{I}_i, \mathrm{I}_{i+1})$. While our proposed method would consider the perturbing pose via the warping operation therefore enabling the network to predict a covariance that varies according to the perturbing pose.

C. Loss Function

We propose to utilise the log-likelihood loss function while assuming a diagonal covariance matrix (departure from the full covariance estimation of previous methods [5][7]), that is,

$$\underset{R_{1:N}}{\arg\min} \sum_{i=1}^{N} -\log(p(e_i|R_i)), \tag{6}$$

where $R_i = D(exp(d_i))$, $d_i \in \mathbb{R}^6$ is output of the network and $D(\cdot)$ maps a vector to a diagonal matrix. Given a diagonal covariance matrix the evaluated loss function is given by,

$$\underset{d_{1:N}}{\arg\min} \sum_{i=1}^{N} \mathrm{sum}(d_i) + e_i^T R_i^{-1} e_i. \tag{7}$$

Our independence assumption on motion parameters alleviates the following problems, 1) a simplified loss function that is more stable during training (no need to compute matrix decomposition's of covariance), and 2) given that we are assuming independently distributed but not identically distributed measurements, we reduce the mismatch between degrees of freedom (d.f.) between the covariance and measurement (i.e. both have six d.f.).

V. Experiments

A. Training Details

Our proposed model includes a depth estimation network along with a covariance estimation network. For the ***depth estimation network*** we propose to utilise the pre-trained monocular depth network proposed in [11] and more specifically the 1024x320 version. The model was trained using the KITTI dataset using the *Eigen* split [15] (we will discuss this point further when we present our KITTI data split) and we do not update the weights during our training. We compute a scale factor (as described in [11]) to convert unscaled depth to a metric depth estimate by using the mean radial distance derived from LiDAR measurements. Given that stereo depth estimation models outperform scaled monocular depth estimation models [16], the use of LiDAR measurements for monocular scale estimation is not unreasonable for obtaining metric depth.

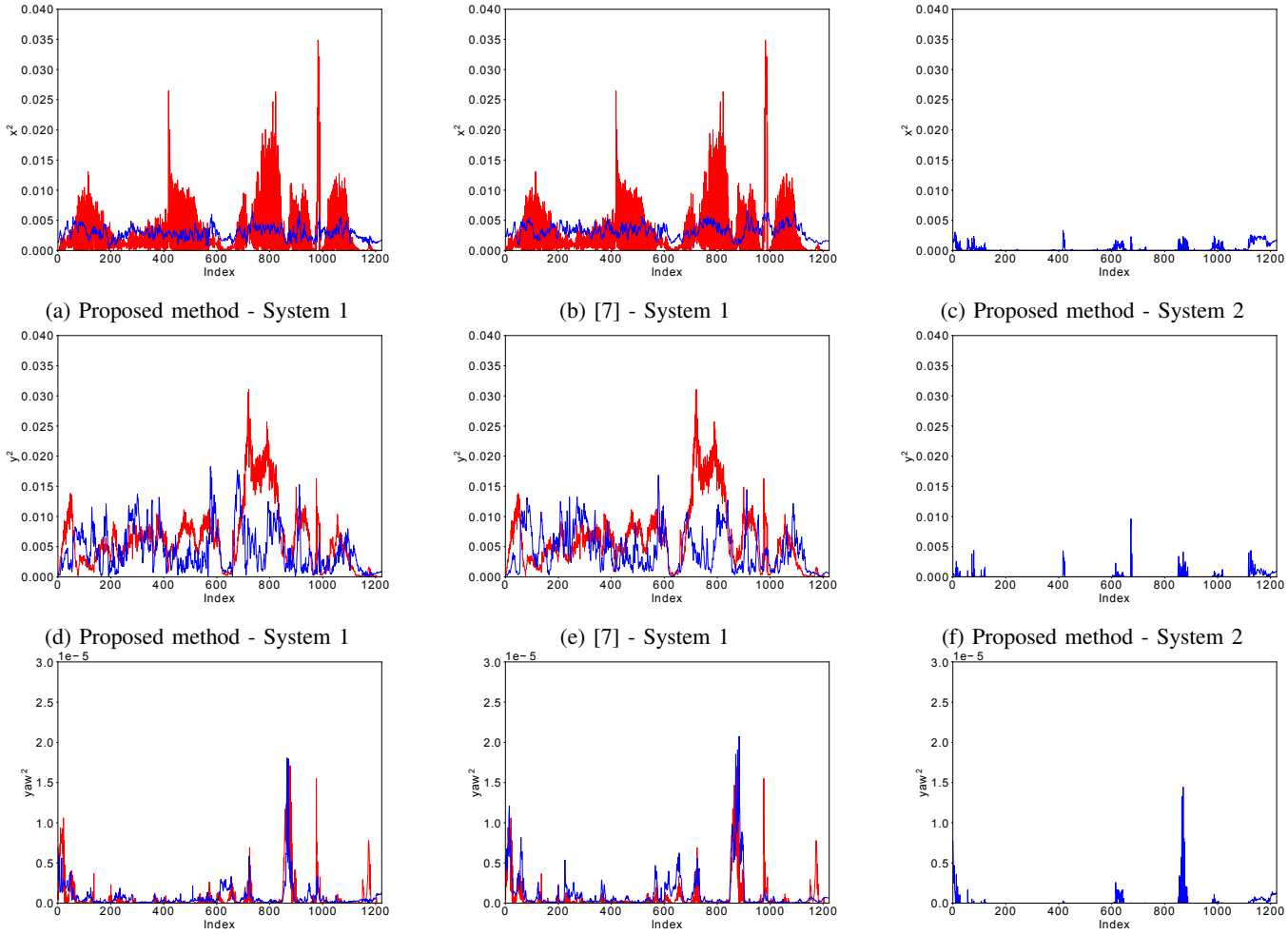
The hyperparameters for ***covariance estimation network*** training were set as follows; batch size: 84, optimiser: Adam, and learning rate: 1e-04. We stopped training according to [7], that is, once we observed diverging train and evaluation losses. Furthermore, similar to [5] we applied a dropout layer at the output of each convolutional layer with dropout rate of 50%. Finally, all experiments were carried out using Nvidia GeForce RTX 3080Ti.

B. Experimental Setup

1) Dataset: We evaluate the proposed method on the KITTI dataset as it contains a diverse range of driving scenes [14]. Following the work in [7] we use the following sequences for training and testing: 04-10; where the train and

Layer	Kernel Size	Stride	No. of outputs
c_1	5x5	2	64
c_2	5x5	2	128
c_3	3x3	2	256
c_4	3x3	2	512
c_5	3x3	1	1024
fc_1	-	-	128
fc_2	-	-	6

TABLE I: Network architecture based on the model in [7].



(a) Proposed method - System 1 (b) [7] - System 1 (c) Proposed method - System 2

(d) Proposed method - System 1 (e) [7] - System 1 (f) Proposed method - System 2

(g) Proposed method - System 1 (h) [7] - System 1 (i) Proposed method - System 2

Fig. 3: The estimated covariance (blue line) overlaid onto the squared error (red line) for the following motion components: the translation covariance along x-axis (top row), the translation covariance along y-axis (middle row), and the yaw rotation covariance (bottom row). Additionally, the predicted covariance corresponding to system 1 relative pose measurements are shown in the left and middle columns, while predicted covariance corresponding to system 2 measurements are shown in the right column. Finally, note that our model is able to process relative pose measurements acquired from different systems.

evaluation/test split follow the work in [11], where sequence 10 is used for evaluation/testing ($N = 1,223$) and all other sequences are used for training ($N = 12,016$). Finally, the *Eigen* split includes a subset of images from sequence 10 in its train set. We do not consider this to be an issue as the depth accuracy (root mean square error) for *Eigen* train (3.28m) set is almost equal to that of *Eigen* evaluation set (3.38m) for KITTI sequence 10.

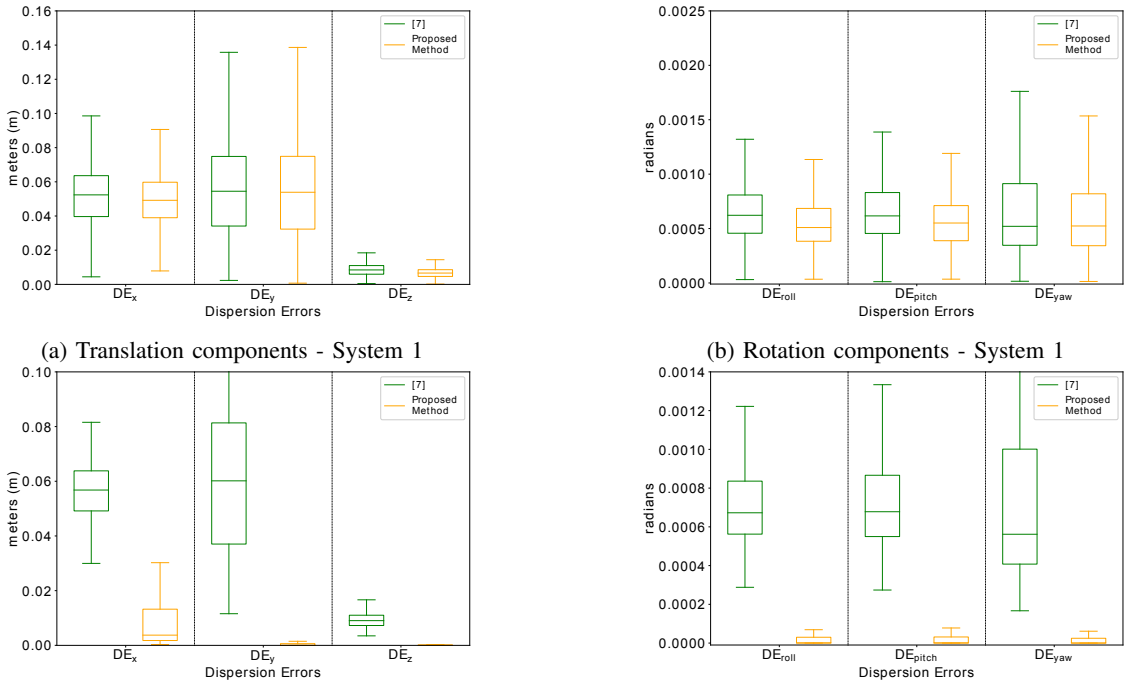
2) Sensor measurements: We seek to demonstrate the capability of our model in estimating covariance for an arbitrary relative pose measurement. To this end, we propose to utilise the following systems for computing relative measurement,

- *System 1*: Relative pose measurement derived from sparse visual odometry techniques (in particular, feature matching based sparse visual odometry [8]), ${}^{i}\hat{\mathrm{T}}_{i+1}^{VO}$.

- *System 2*: Relative pose measurements, ${}^{i}\hat{\mathrm{T}}_{i+1}^{GNSS}$, derived from GNSS, where the objective of including this measurement is to demonstrate that our method should be able to predict covariance for low uncertainty measurements. We also consider the ground truth (${}^{i}\mathrm{T}_{i+1} = {}^{i}\hat{\mathrm{T}}_{i+1}^{GNSS}$) to be GNSS relative pose.

For each relative pose measuring system we also have the same image pairs I_i and I_{i+1}; that is, for the same pair of viewpoints, the error distribution of the relative pose measurement varies according to the system. Our model was trained using system 1 and system 2 measurements, where for each mini-batch during training, we maintain 75% of the data from system 1, and 25% from system 2; owing to the disproportionately high impact of system 2 measurements (zero squared error) on the loss as compared with system 1 measurements. Finally, the method in [7] was trained using system 1 measurements.

3) Evaluation metrics: We use the **median dispersion error (MDE)** (a variation of median absolute error) to



(a) Translation components - System 1

(b) Rotation components - System 1

(c) Translation components - System 2

(d) Rotation components - System 2

Fig. 4: Boxplots of the dispersion errors (that is, error between predicted covariance and ground truth relative pose error) for translation and rotation motion components. Our proposed model is able to process system 1 (top row) and system 2 (bottom row) relative pose measurements in order to predict a covariance. For reference, we have included results of the predicted covariance evaluated against the ground truth errors of system 2 measurements using the method in [7].

quantitatively evaluate our proposed method,

$$MDE_c = median\left(\sqrt{|R_{i,c} - e_{i,c}^2|}\right) \quad \text{for} \quad i = 1, .., N$$

where c corresponds to each motion component (e.g. x-translation), while $R_{i,c}$ correspond to the diagonal element of the covariance corresponding to each motion component. Finally, given the statistical assumptions used in this work (i.e. identically distributed errors), we assume that ground truth of covariance is given by e_i^2.

Rel. Pose. Meas.	Metric	Proposed Model	[7]
System 1	MDE_x	**0.049**	0.052
	MDE_y	0.054	0.054
	MDE_z	**0.0066**	0.0084
	MDE_{roll}	**5.0e-04**	6.3e-04
	MDE_{pitch}	**5.5e-04**	6.1e-04
	MDE_{yaw}	5.2e-04	**5.1e-04**
System 2	MDE_x	0.0037	0.056
	MDE_y	2.76e-06	0.060
	MDE_z	7.17e-07	0.0090
	MDE_{roll}	1.04e-06	5.0e-4
	MDE_{pitch}	1.05e-06	5.5e-4
	MDE_{yaw}	2.46e-07	5.2e-4

TABLE II: MDE_c computed for all motion components (translations and rotations) for both the proposed method and [7], along with the relative pose measurements acquired from both system 1 and system 2.

C. Experimental Analysis

In this section we analyse the performance of our proposed method using the evaluation dataset. In particular, we directly compare the performance of our proposed method with [7] for system 1 measurements (as [7] can only be trained on relative pose measurements acquired from a single system), while simultaneously highlighting the performance of our model in processing system 2 relative pose measurements. The rows in Fig. 3 show the output of the predicted covariance for the following motion components: x-translation (top row), y-translation (middle row) and yaw-rotation (bottom row); while the columns in Fig. 3 show the following methods: proposed model processing system 1 measurements (left column), [7] processing system 1 measurements (middle column), and proposed model processing system 2 measurements (right column). From Fig. 3 we can qualitatively assess that covariance predicted for system 1 measurements using our proposed method (Fig. 3 (a),(d),(g)) is similar to the method in [7] (Fig. 3 (b),(e),(h)). If the relative pose measurement is acquired from system 2, we can observe that our proposed model is able to vary the predicted covariance (Fig. 3 (c),(f),(i)). It should be noted that our proposed model generated spurious artifacts when predicting covariance for system 2 measurements. We do not know the exact cause, but we speculate it may be related to the size of the training data set.

Table. II shows the median dispersion errors for all translation and rotation components for the respective methods

and systems. Additionally Fig. 4 shows the corresponding box plots of the dispersion errors (DE_c). For system 1 measurements, both the MDE scores (shown in Table. II) and the box plots of the dispersion errors (top row Fig. 4) are similar. While for system 2 measurements, it can be observed the proposed method is able to predict covariance that is close to the ground truth squared error, demonstrating the potential for our system to adapt the predicted covariance according to the relative pose measurement system (for reference we have included MDE scores and corresponding boxplots for the predicted covariances of system 2 measurements using [7]).

VI. CONCLUSIONS

In this work we have proposed a covariance estimation method for relative pose measurements obtained from an arbitrary system. We achieve this by utilising the measured relative pose along with the images acquired between the two viewpoints in order to compute a covariance. We demonstrate the ability of our model to estimate covariance for relative pose measurements obtained from two different systems, while existing state of the art methods are limited to measurements obtained from a single system. Finally, in future work, we aim to investigate viewpoint synthesis via NeRFs [17] as replacement to image warping, along with relative pose measurement generation to augment existing real world datasets.

REFERENCES

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

[2] J. Ziegler, H. Lategahn, M. Schreiber, C. G. Keller, C. Knöppel, J. Hipp, M. Haueis, and C. Stiller, "Video based localization for Bertha," *IEEE Intelligent Vehicles Symposium*, pp. 1231-1238, 2014.

[3] D. Caruso, J. Engel, and D. Cremers, "Large-scale direct SLAM for omnidirectional cameras," *IEEE International Conference on Intelligent Robot Systems*, 2015.

[4] A. Hardt-Stremayr and S. Weiss, "Monocular visual-inertial odometry in low-textured environments with smooth gradients: A fully dense direct filtering approach," *IEEE International Conference on Robotics and Automation*, pp. 7837-7843, 2020.

[5] K. Liu, K. Ok, W. Vega-Brown and N. Roy, "Deep inference for covariance estimation: Learning Gaussian noise models for state estimation," *IEEE International Conference on Robotics and Automation*, pp. 1436-1443, 2018.

[6] H. Hu and G. Kantor, "Parametric covariance prediction for heteroscedastic noise," *IEEE International Conference on Intelligent Robots and Systems*, pp. 3052-3057, 2015.

[7] A. de Maio and S. Lacroix, "Simultaneously learning corrections and error models for geometry-based visual odometry methods," *IEEE Robotics and Automation Letters*, vol. 5 , no. 4, pp.6536–6543, 2020.

[8] D. Scaramuzza and F. Fraundorfer, "Visual odometry [tutorial]," *IEEE robotics and automation magazine*, vol. 18, no. 4, pp. 80–92, 2011.

[9] H. Strasdat, J. M. M. Montiel and A. J. Davison, "Scale drift-aware large scale monocular SLAM," *Robotics: Science and Systems (RSS)*, 2010.

[10] M. Jaderberg, K. Simonyan, A. Zisserman and K. Kavukcuoglu, "Spatial transformer networks," *In NeurIPS*, 2015.

[11] M. Brossard, A. Barrau and S. Bonnabel, "AI-IMU Dead-Reckoning," *IEEE Transactions on Intelligent Vehicles*, vol. 5, no. 4, pp. 585–595, 2020.

[12] T. Zhou, M. Brown, N. Snavely and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," IEEE Conference on Computer Vision and Pattern Recognition, pp. 6612-6619, 2017.

[13] C. Godard, O. M. Aodha, M. Firman and G. Brostow, "Digging into self-supervised monocular depth estimation," *IEEE International Conference on Computer Vision*, pp. 3827-3837, 2019.

[14] A. Geiger, P. Lenz, C. Stiller and R. Urtasun, "Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research*. vol. 32, no. 11, pp. 1231–1237, 2013.

[15] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," *In NeurIPS*, pp. 2366–2374, 2014.

[16] Z. Chen, X. Ye, W. Yang, Z. Xu, X. Tan, Z. Zou, E. Ding, X. Zhang and L. Huang, "Revealing the reciprocal relations between self-supervised stereo and monocular depth estimation," *IEEE International Conference on Computer Vision*, pp. 15509–15518, 2021.

[17] C. Feldmann, N. Siegenheim, N. Hars, L. Rabuzin, M. Ertugrul, L. Wolfart, M. Pollefeys, Z. Bauer and M. R. Oswald, "NeRFmentation: NeRF-based augmentation for monocular depth estimation," *arXiv:2401.03771* [cs], 2024.