Learned Improvements to the Visual Egomotion Pipeline

Final Oral Examination

Valentin Peretroukhin

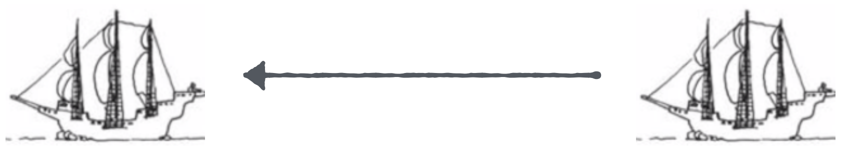
Supervised by Professor Jonathan Kelly

Institute for Aerospace Studies
UNIVERSITY OF TORONTO

STARS LABORATORY

Egomotion Estimation

and 'Dead' Reckoning



Egomotion estimation: the process of estimating the motion of a rigid body using measurements from sensors attached to the body

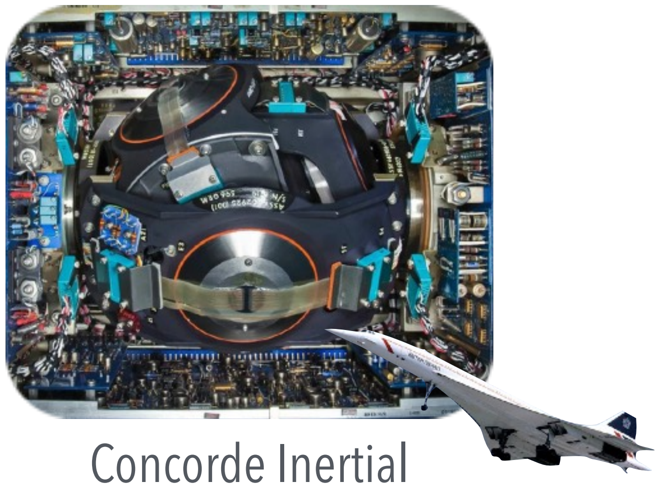
- Methods of egomotion estimation:

  1. exteroceptive measurements
     observing landmarks with known location

  2. interoceptive measurements
     integrating rates to infer motion through '**dead' reckoning**

Charles Lindbergh used dead reckoning to cross the Atlantic solo in 1927

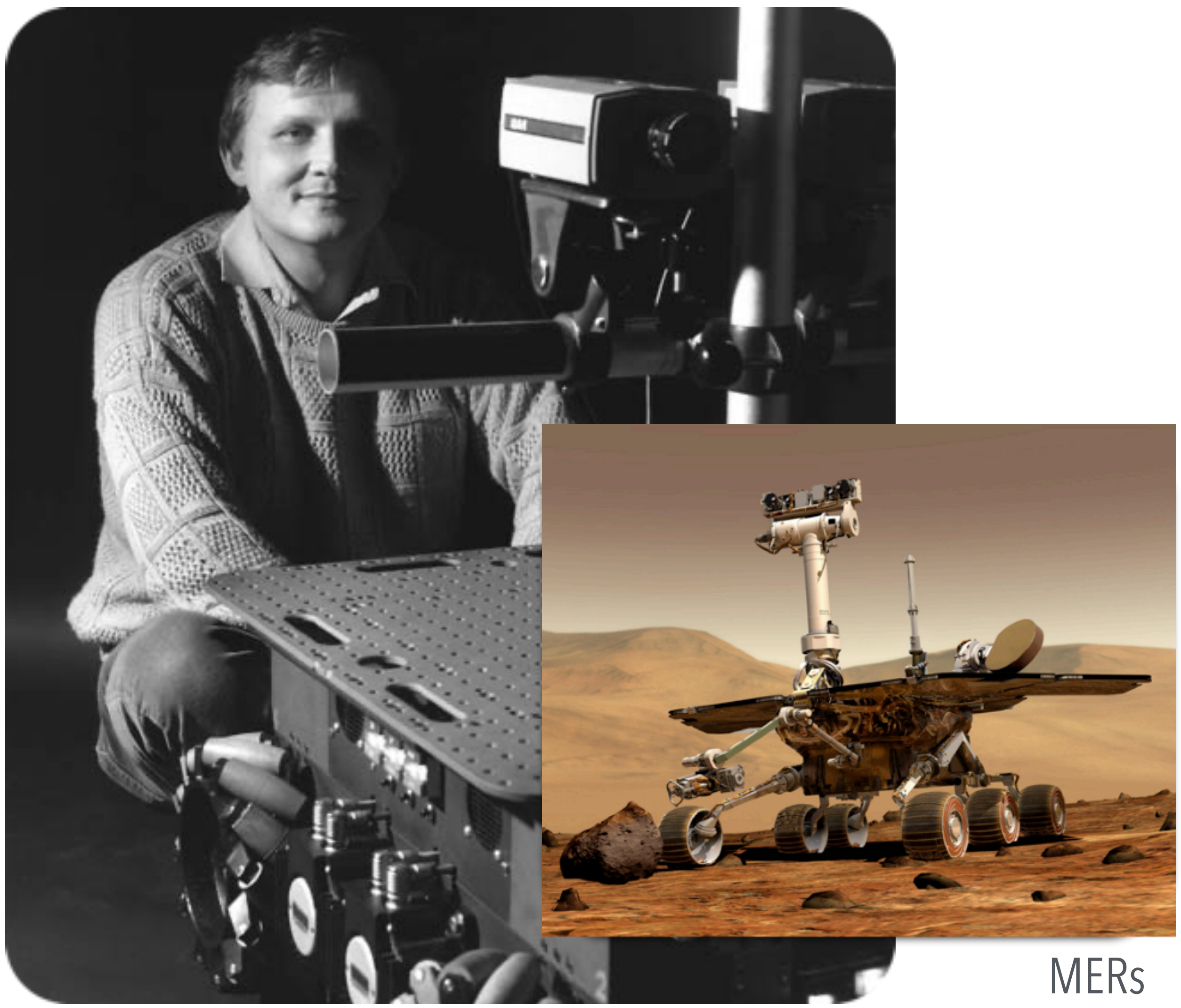
Apollo Delco (GM) IMU



Concorde Inertial Measurement Unit

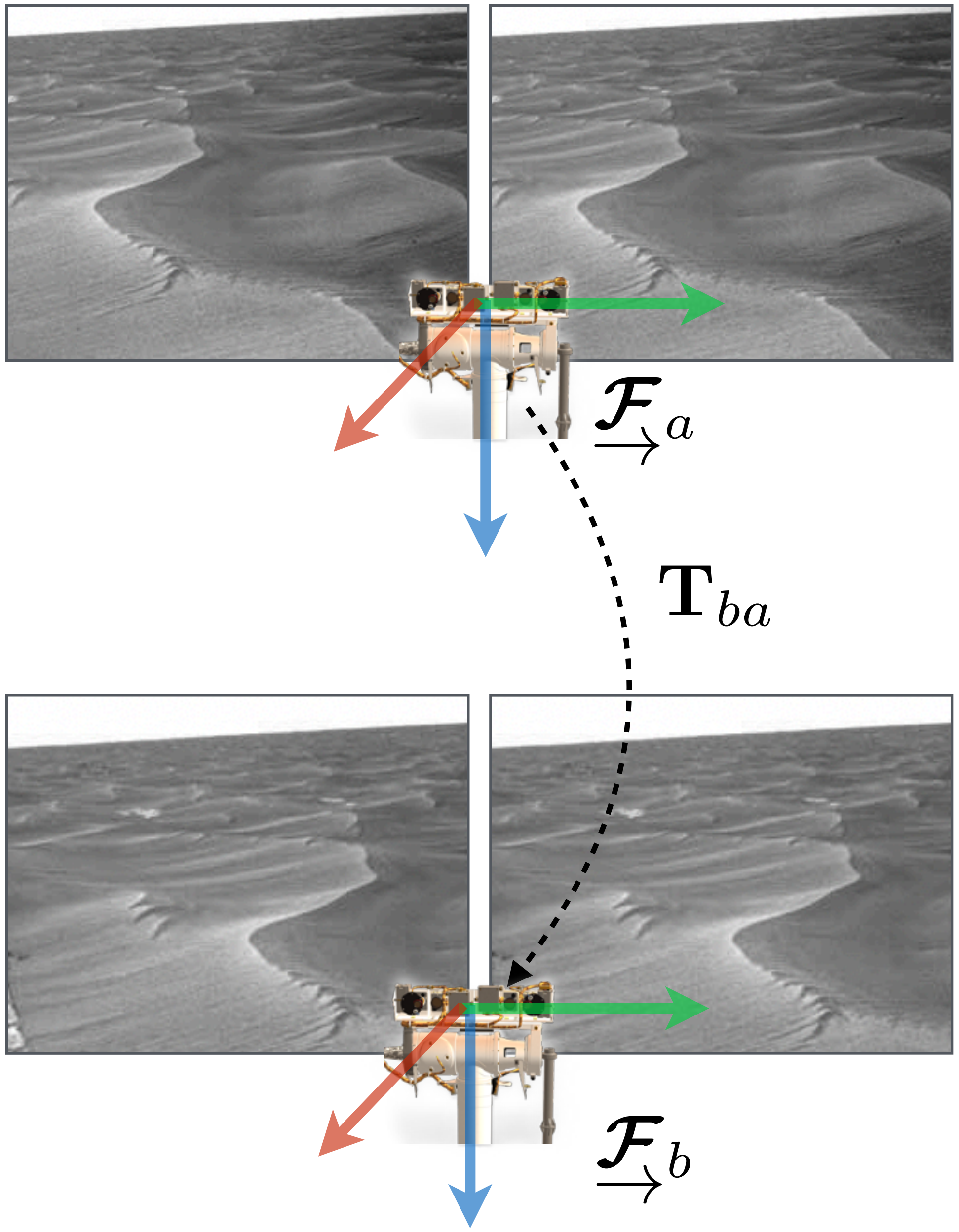
Visual Egomotion Estimation

or Visual *Odometry*



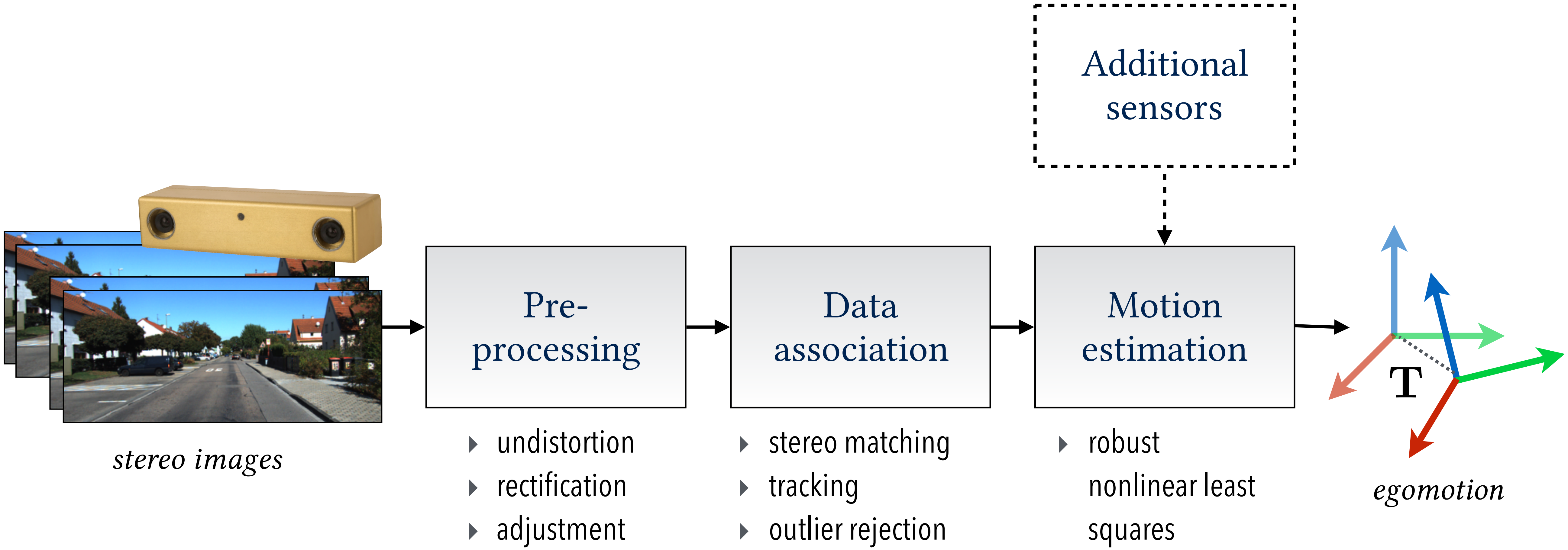
MERs

Hans Moravec



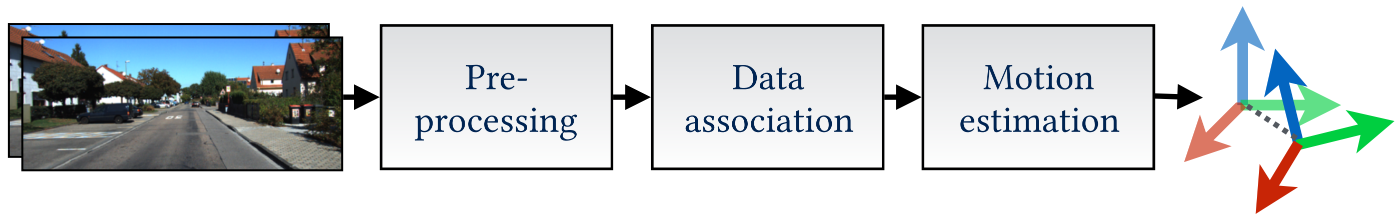
Scaramuzza and Fraundorfer, "**Visual Odometry [Tutorial],**" **IEEE Robot. Automat. Mag.** (2011)
Moravec, "**Obstacle avoidance and navigation in the real world by a seeing robot rover,**" **Ph.D. Thesis** (1980)

Visual Egomotion *Pipeline*

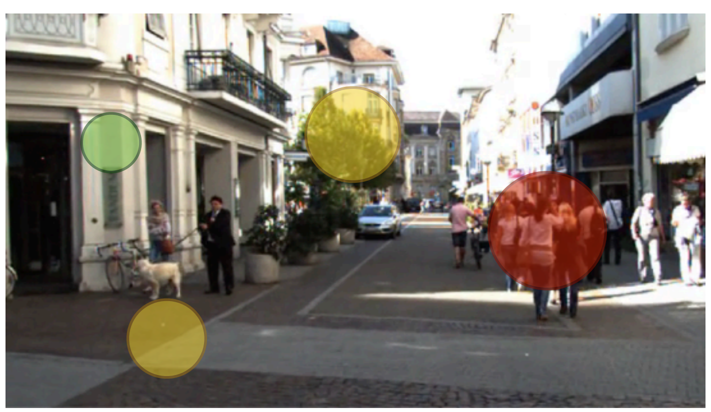


Visual Egomotion Pipelines

Some Downsides…

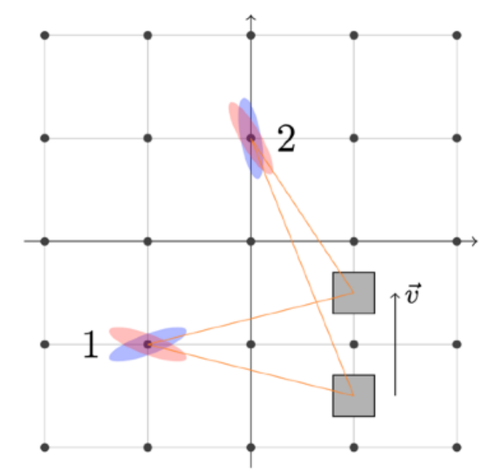


Basic Uncertainty Quantification
(e.g., homoscedastic isotropic uncertainty)



Low Information Usage
(e.g., point features, regions of high gradients)

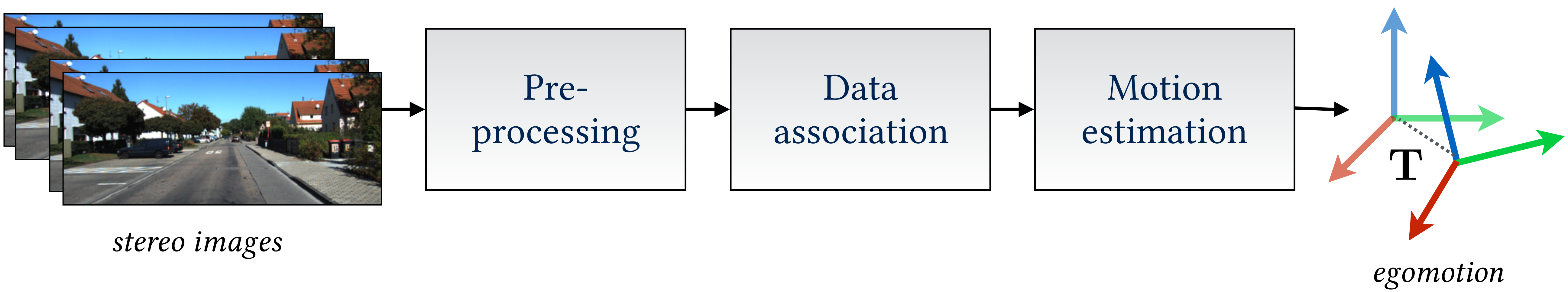
Prone to Bias
(e.g., imprecise calibration, uncertainty propagation)



Peretroukhin, Kelly, and Barfoot, "**Optimizing Camera Perspective for Stereo Visual Odometry**," **CRV** (2014)

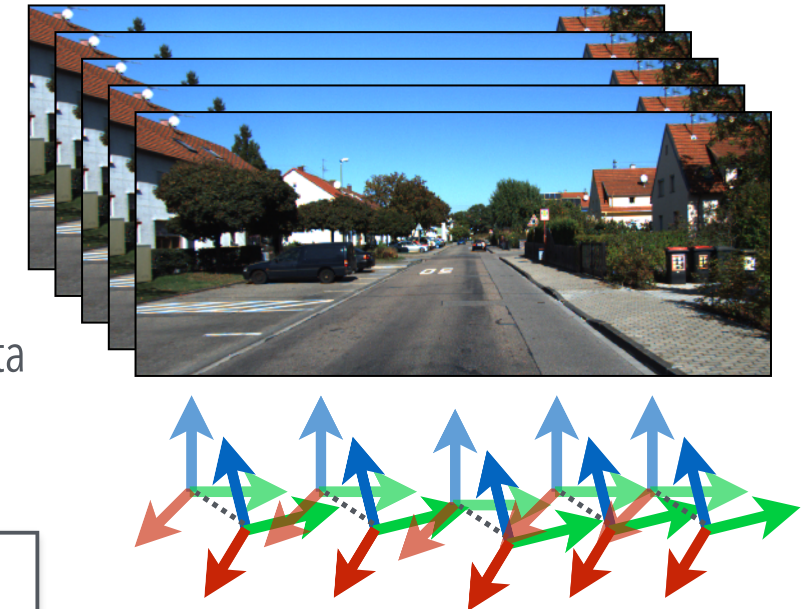
Data-Driven Learning

A Panacea for Vision-based Autonomy?



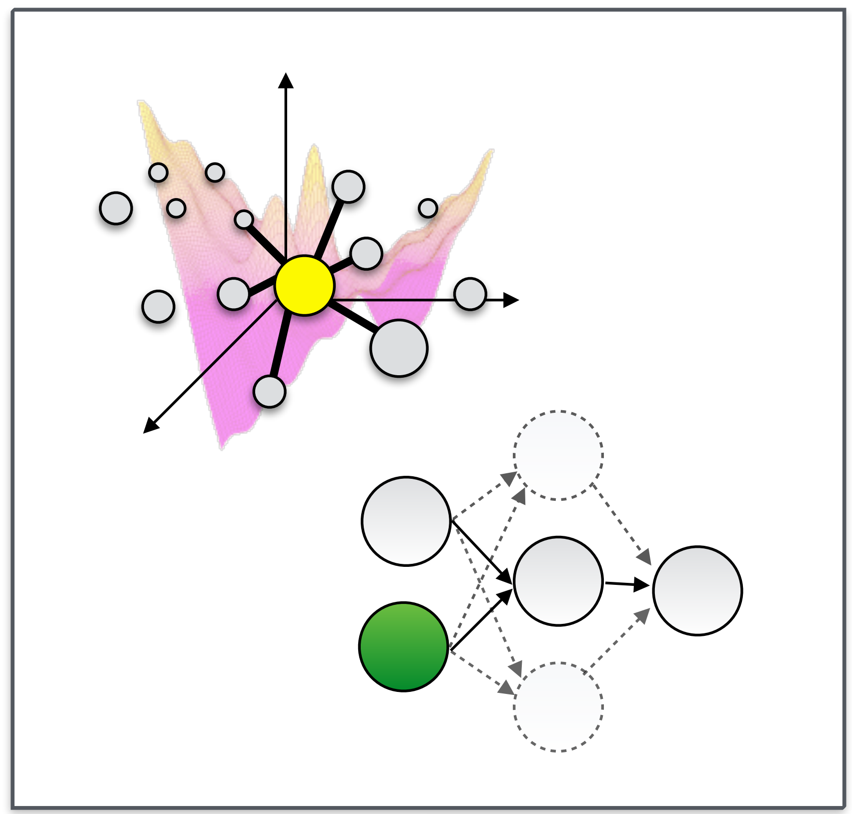
Data-Driven Learning

A Panacea for Vision-based Autonomy?

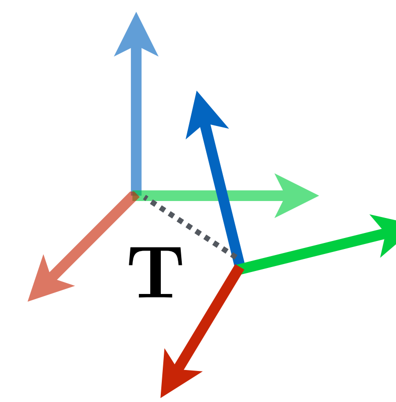


Training Data

stereo images



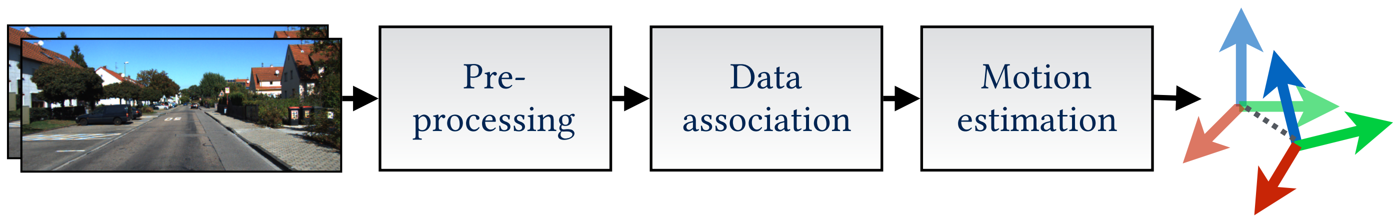
Learned Model



egomotion

Wang et al., "DeepVO: Towards end-to-end visual odometry with deep Recurrent Convolutional Neural Networks," **ICRA** (2017)
Zhou et al, "Unsupervised Learning of Depth and Ego-Motion from Video," **CVPR** (2017)

Benefits of Classical Pipelines



Interpretable & decomposable

Zhou et al. "**Does computer vision matter for action?**", **Science Robotics** (2019)

Short answer: **Yes!** intermediate 'blocks' improve generalization

Efficient

Probabilistic

Accurate

KITTI Odometry Benchmark Leaderboard (March 2020)

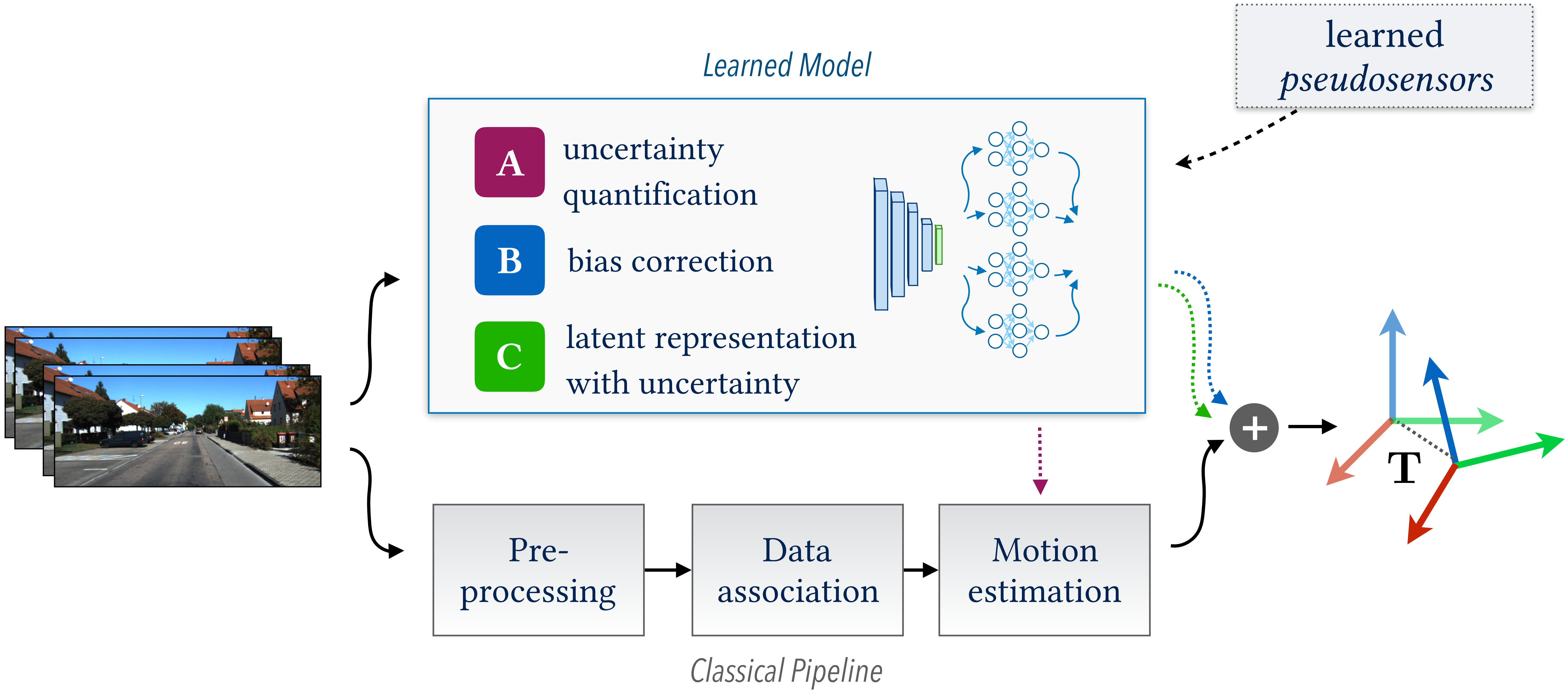
	Method	Setting	Code	Translation	Rotation	Runtime	Environment	Compare
1	V-LOAM			0.55 %	0.0013 [deg/m]	0.1 s	2 cores @ 2.5 Ghz (C/C++)	
J. Zhang and S. Singh: Visual-lidar Odometry and Mapping: Low drift, Robust, and Fast. IEEE International Conference on Robotics and Automation(ICRA) 2015.								
2	LOAM			0.57 %	0.0013 [deg/m]	0.1 s	2 cores @ 2.5 Ghz (C/C++)	
J. Zhang and S. Singh: LOAM: Lidar Odometry and Mapping in Real- time. Robotics: Science and Systems Conference (RSS) 2014.								
3	SOFT2			0.65 %	0.0014 [deg/m]	0.1 s	2 cores @ 2.5 Ghz (C/C++)	
I. Cvišić, J. Ćesić, I. Marković and I. Petrović: SOFT-SLAM: Computationally Efficient Stereo Visual SLAM for Autonomous UAVs. Journal of Field Robotics 2017.								

Lidar-based Egomotion

top performing ***vision-only*** odometry has ***no learning***

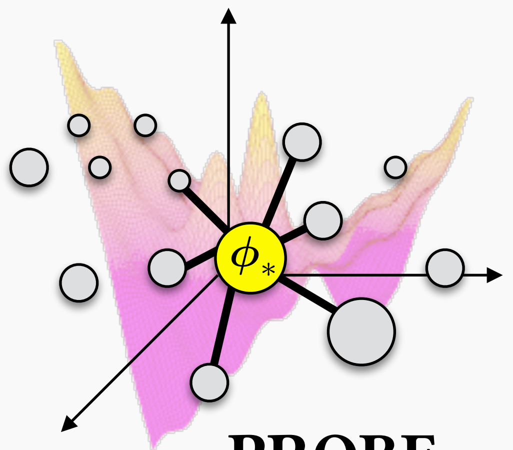
My Doctoral Work

Learned Improvements to the Visual Egomotion Pipeline



Learned Improvements

I uncertainty quantification



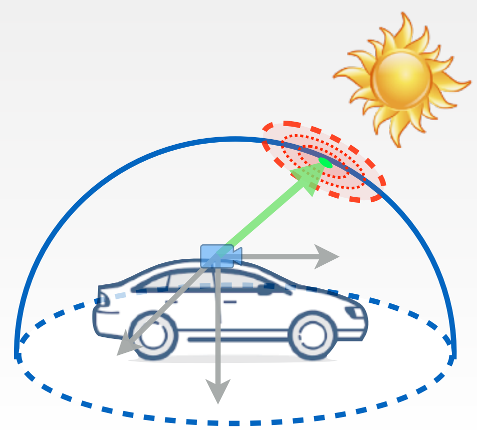
PROBE

Predictive Robust Estimation

IROS 2015

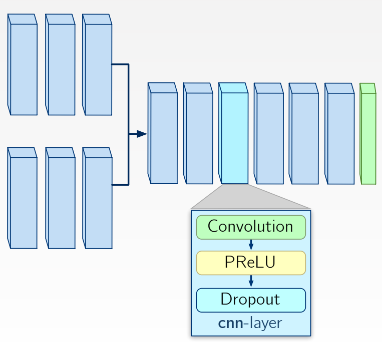
ICRA 2016

II latent representation



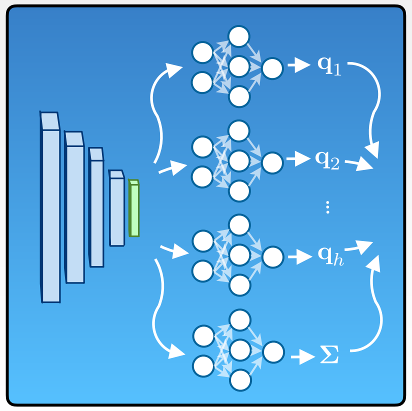
Sun-**BCNN**

III bias correction



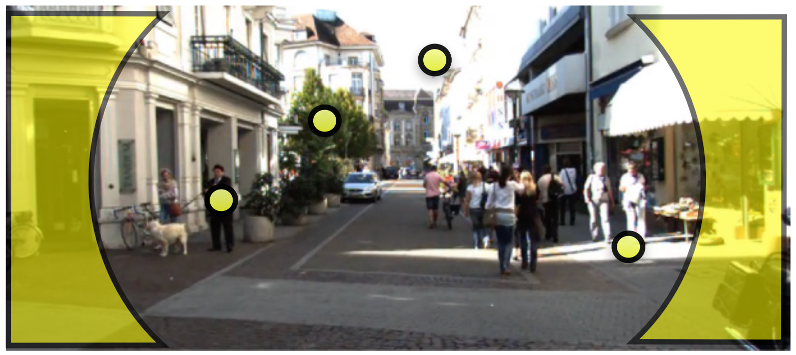
DPC-Net

IV latent representation

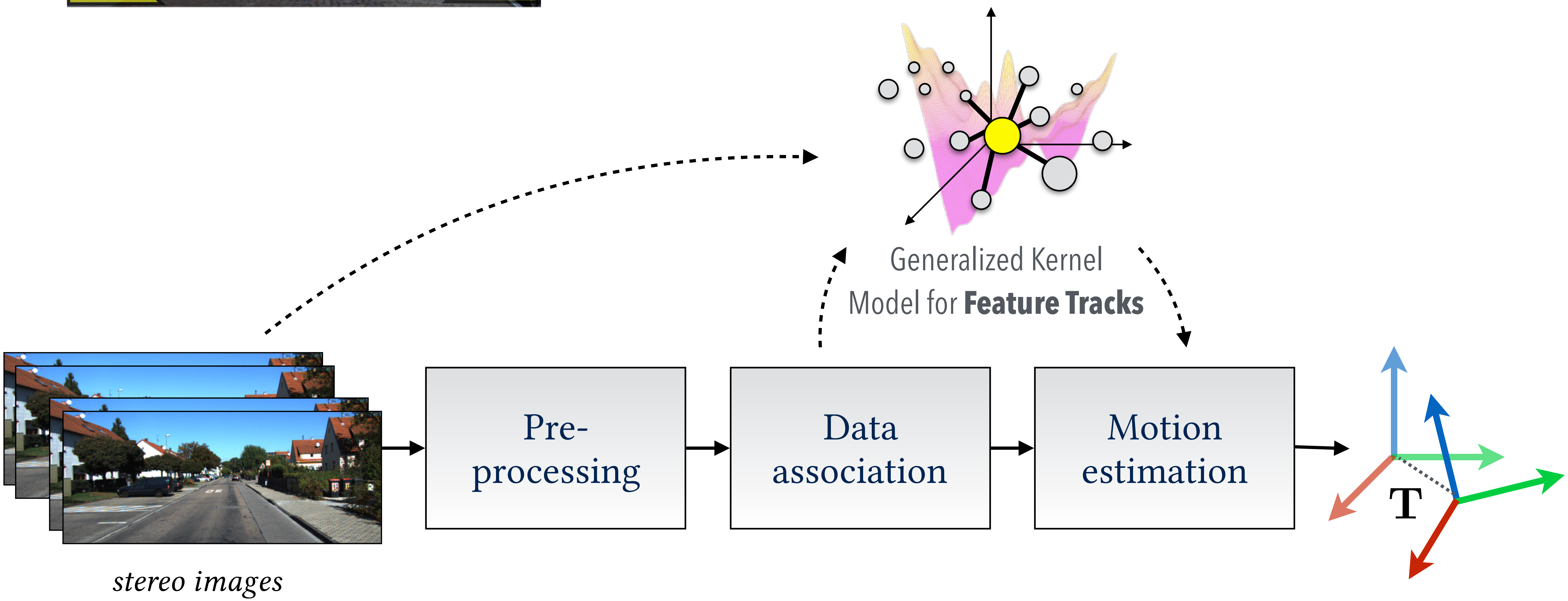


HydraNet

Predictive Robust Estimation (PROBE)



Not all feature matches contain the same information. Can we incorporate a **Bayesian model for stereo reprojection errors** into an egomotion pipeline?



W. R. Vega-Brown, M. Doniec, and N. G. Roy, “**Nonparametric Bayesian inference on multivariate exponential families**,” **NeurIPS** (2014)
V. Peretroukhin, W. Vega-Brown, N. Roy, and J. Kelly, “**PROBE-GK: Predictive Robust Estimation using Generalized Kernels**,” **ICRA** (2016)

Predictive Robust Estimation

Classical Visual Odometry

Errors modelled with **stationary** point uncertainty:

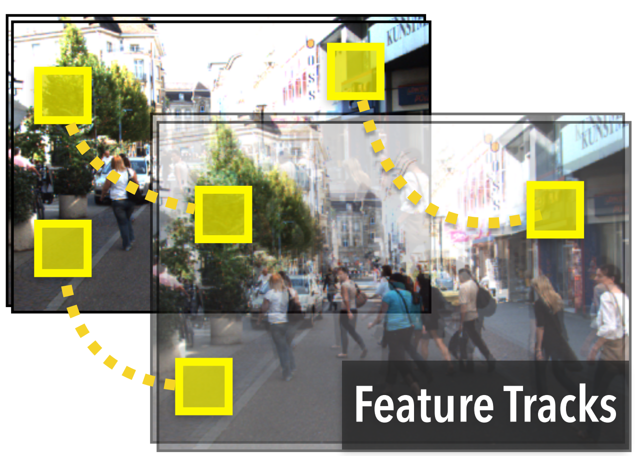
$$\mathbf{e}_i \sim \mathcal{N}\left(\mathbf{0}, \mathbf{R}\right)$$

MAP Estimator

$$\underset{\mathcal{T} \in \mathrm{SE}(3)}{\arg\min} \sum_{i=1}^{N} \boldsymbol{e}_i{}^{\top} \mathbf{R}^{-1} \boldsymbol{e}_i$$

Visual Odometry with

PROBE-GK

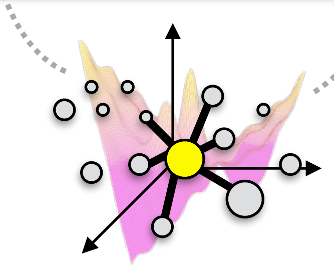


Errors $\mathbf{e}_i$ modelled with **Bayesian,** kernel-based covariance density

$$p\left(\mathbf{R}_i\right) = \mathrm{IW}\left(\mathbf{R}; \boldsymbol{\Psi}_i, \nu_i\right)$$

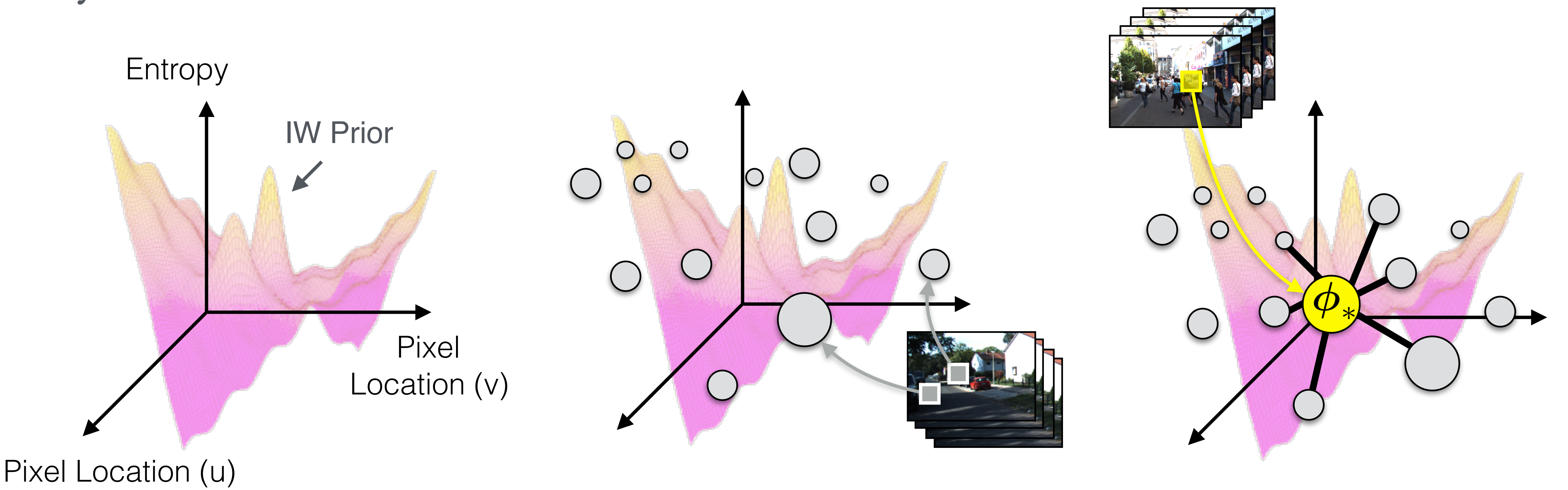
Predictively Robust Estimator

$$\underset{\mathcal{T} \in \mathrm{SE}(3)}{\arg\min} \sum_{i=1}^{N} \left(\nu_i + 1\right) \log\left(1 + \boldsymbol{e}_i{}^{\top} \boldsymbol{\Psi}_i^{-1} \boldsymbol{e}_i\right)$$



Uncertainty Model

Bayesian Predictive Model for Covariance



1. **Define**: Inverse-Wishart prior on covariance matrices at all locations within prediction space ϕ
2. **Train**: Populate space with training data (empirical errors). Ground truth not required.
3. **Test**: a) Compute Inverse-Wishart posterior using technique of Generalized Kernels.

3. b) Marginalize out the covariance, solve robust cost.

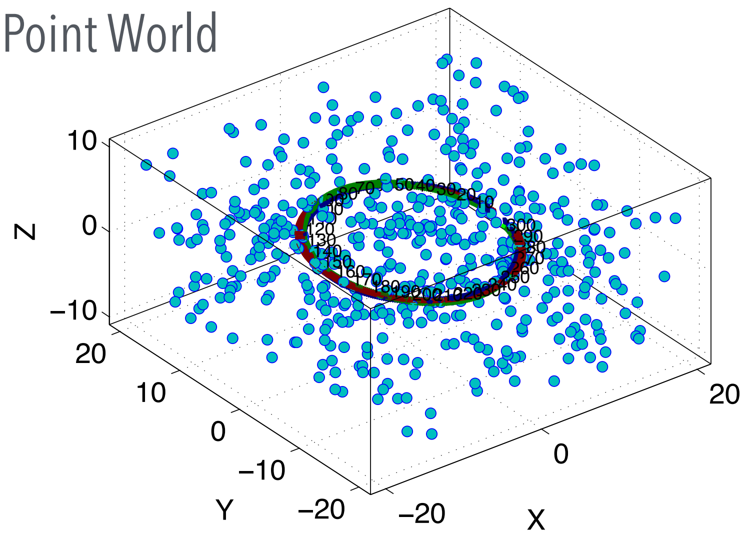
Predictively **Robust** Cost

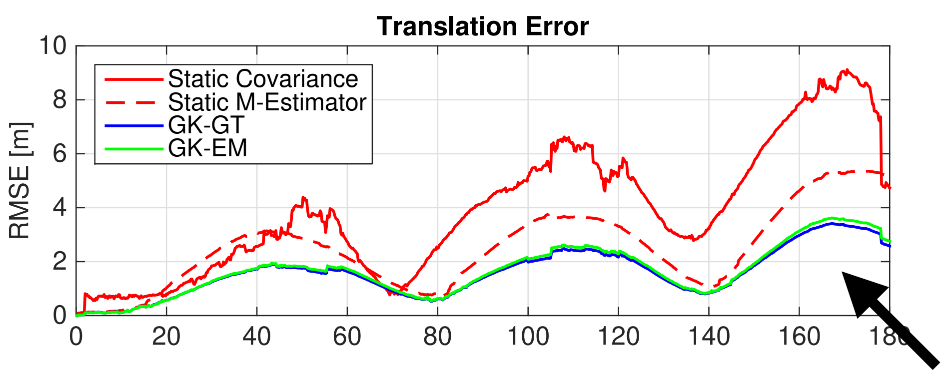
$$\underset{\mathcal{T}\in \mathrm{SE}(3)}{\arg\min} \sum_{i=1}^{N} (\nu_i + 1) \log\left(1 + \boldsymbol{e}_i{}^{\top} \boldsymbol{\Psi}_i^{-1} \boldsymbol{e}_i\right)$$

Using PROBE to Improve VO

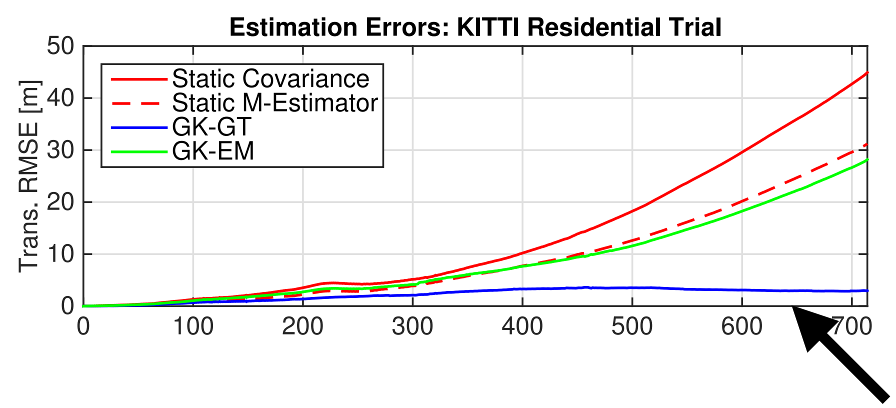
Synthetic, KITTI and UTIAS Experiments

Synthetic Point World

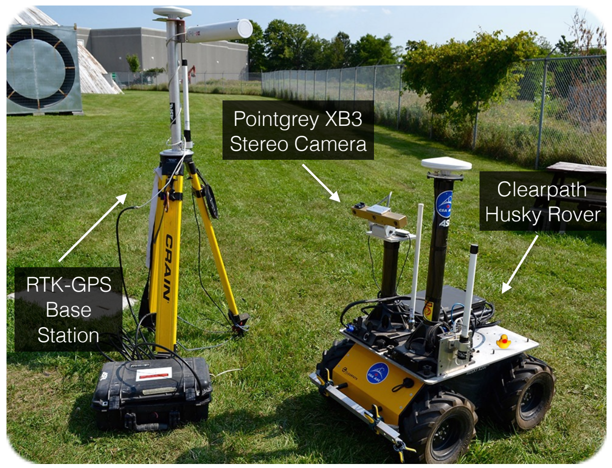


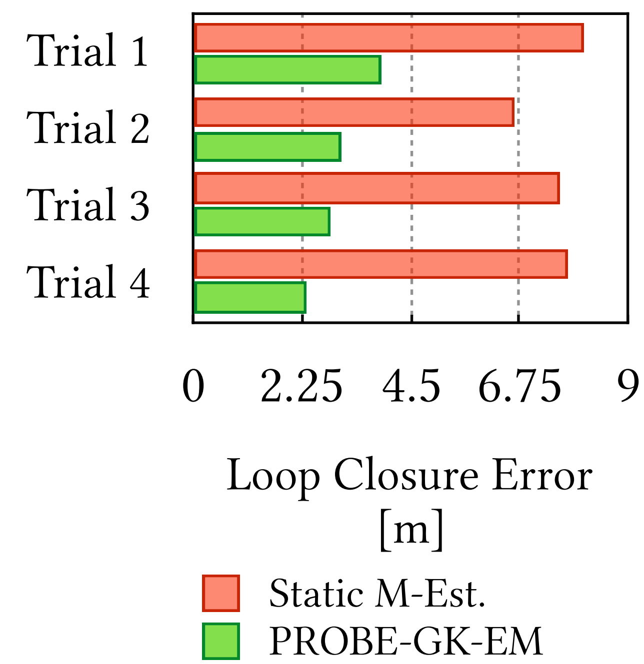


KITI Odometry Dataset



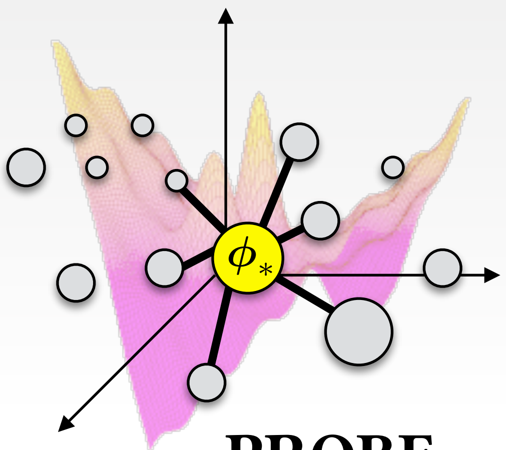
UTIAS Mars Dome





Learned Improvements

I uncertainty quantification

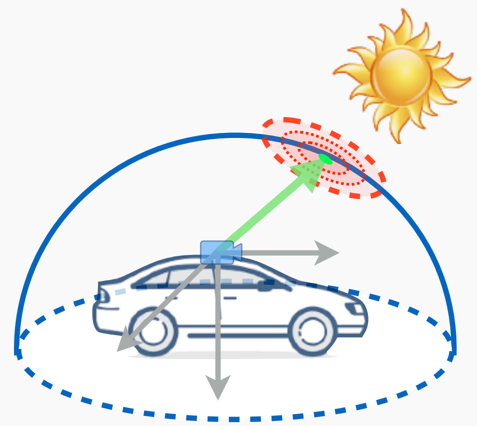


PROBE

Predictive Robust Estimation

IROS 2015
ICRA 2016

II latent representation

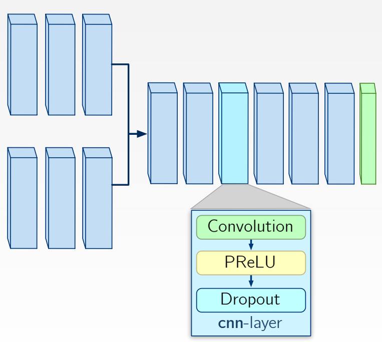


Sun-**BCNN**

Learning Sun Direction with Uncertainty

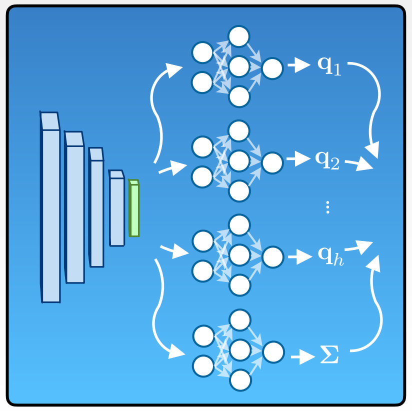
ISER 2017, ICRA 2017, IJRR 2018

III bias correction



DPC-Net

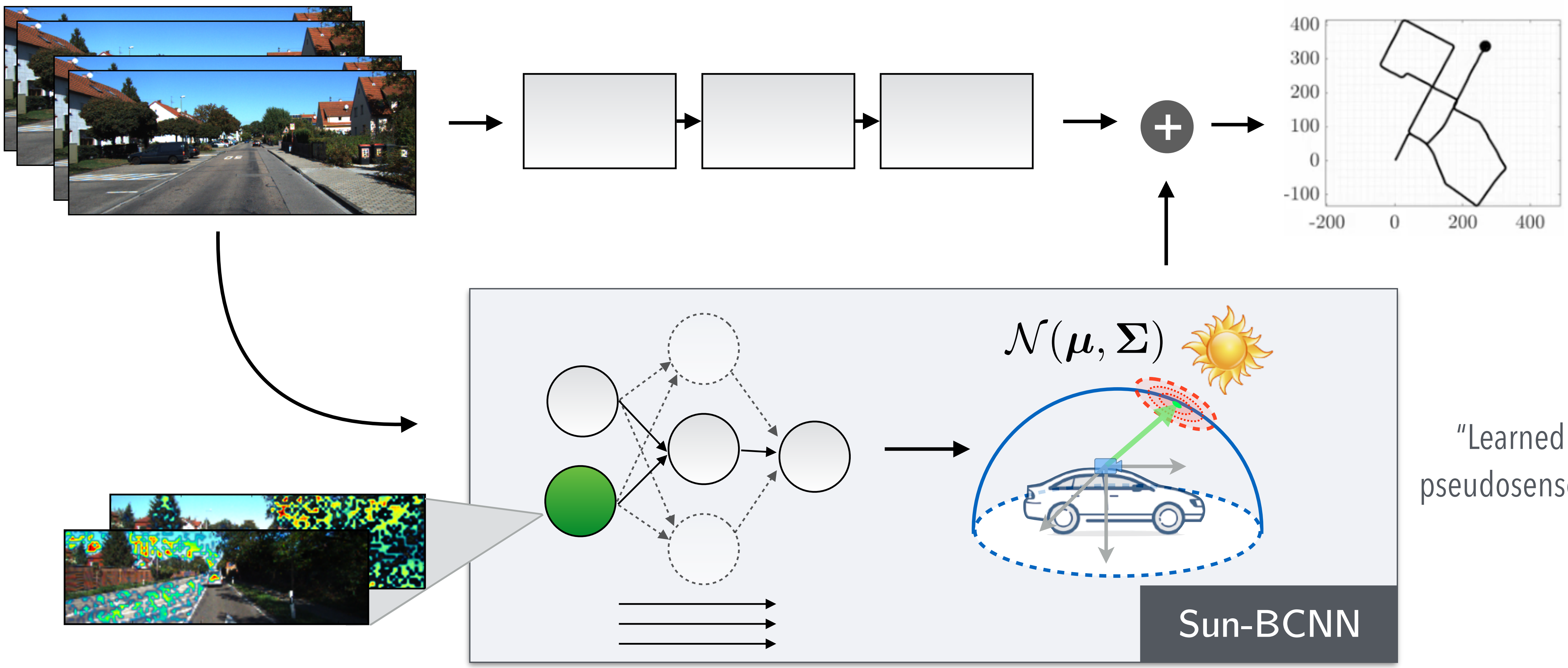
IV latent representation



HydraNet

Sun-BCNN: A Virtual Sun Sensor

Can we use deep learning to infer the direction of the sun (with *uncertainty*) and use it to improve egomotion estimates?

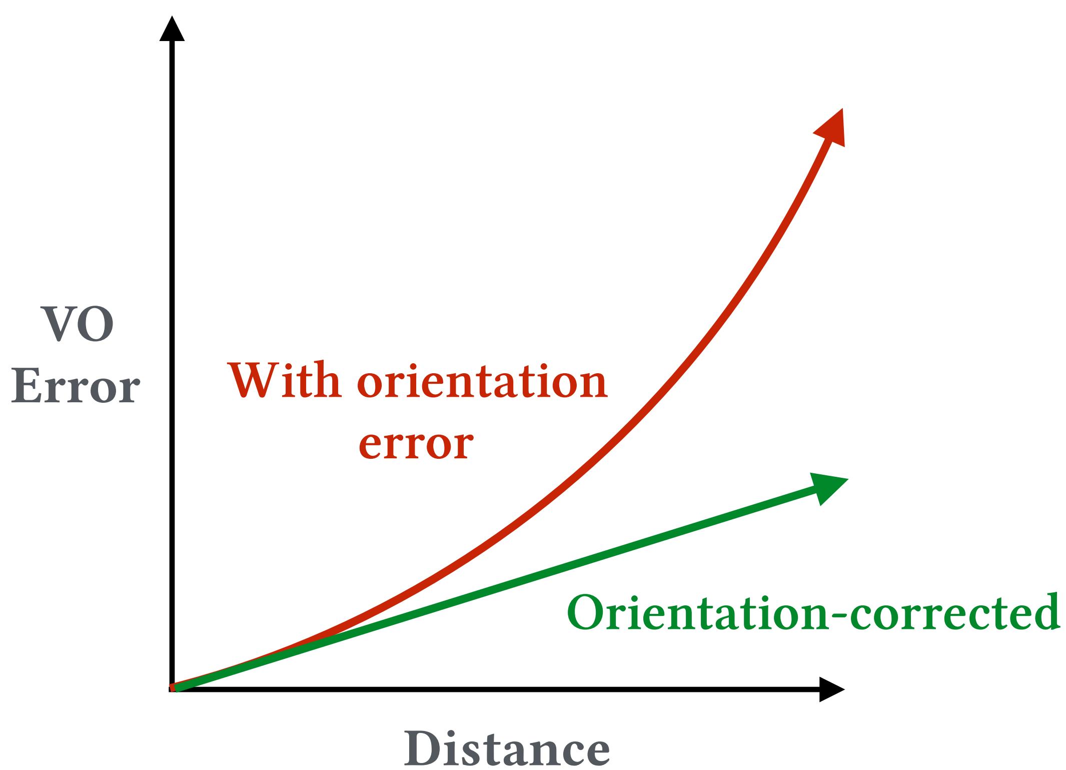


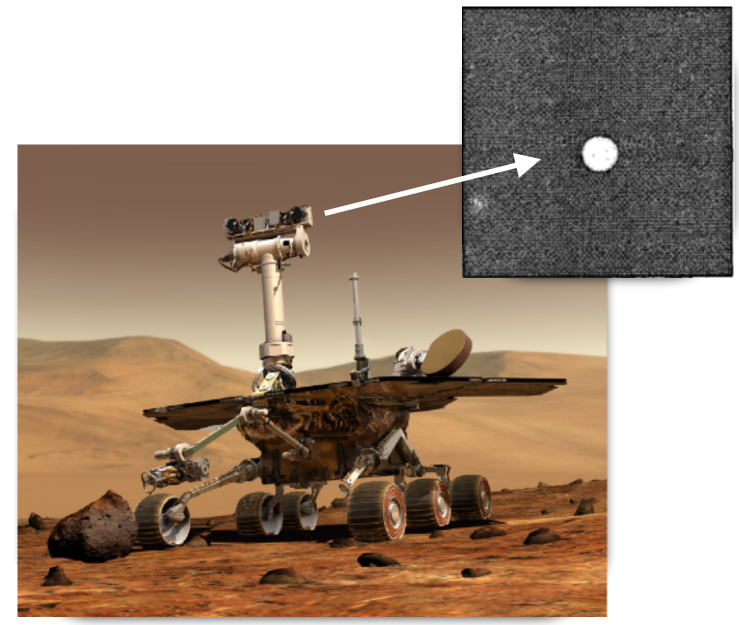
"Learned pseudosensor"

Peretroukhin, Clement, and Kelly, "**Reducing Drift in Visual Odometry by Inferring Sun Direction Using a Bayesian Convolutional Neural Network**," **ICRA** (2017)
Peretroukhin, Clement, and Kelly, "**Inferring sun direction to improve visual odometry: A deep learning approach**," **IJRR** (2018)

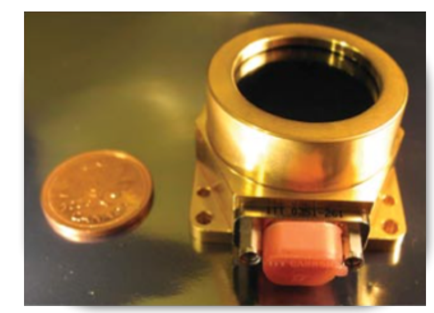
Sun-aided Visual Odometry

Visual odometry is a dead-reckoning technique and suffers from **super-linear error growth**, largely due to accumulated orientation error





Specially oriented camera (e.g., MERs)



Specialized sun sensor

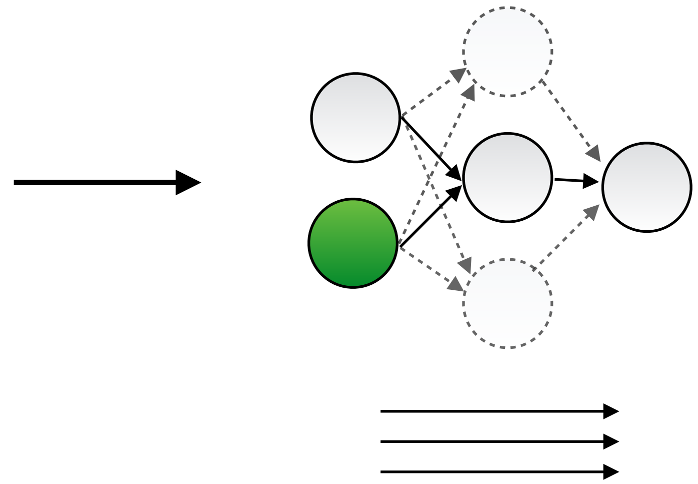
Drift can be reduced using **absolute orientation information** (e.g., observing the Sun)

Sun-BCNN

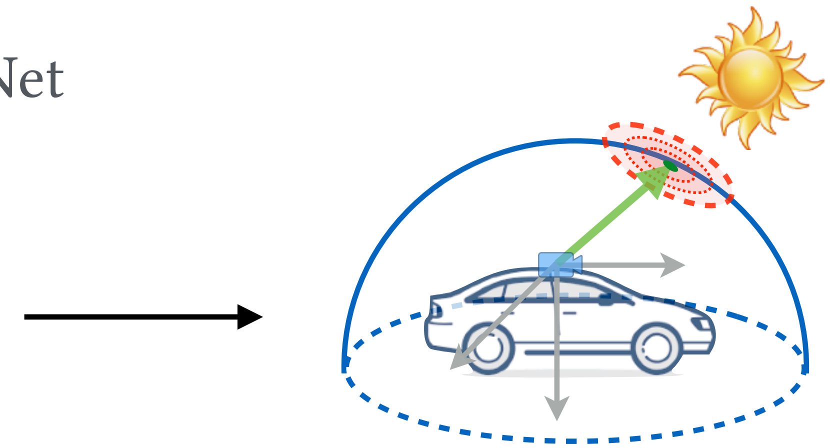
A Bayesian CNN for Finding the Sun

RGB image

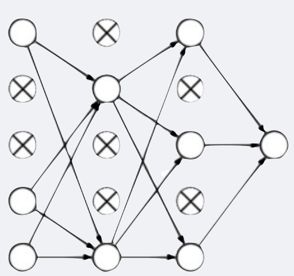
Bayesian GoogLeNet

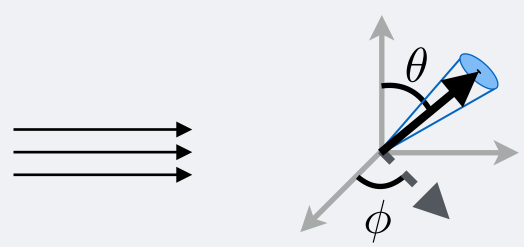


'Monte Carlo' dropout



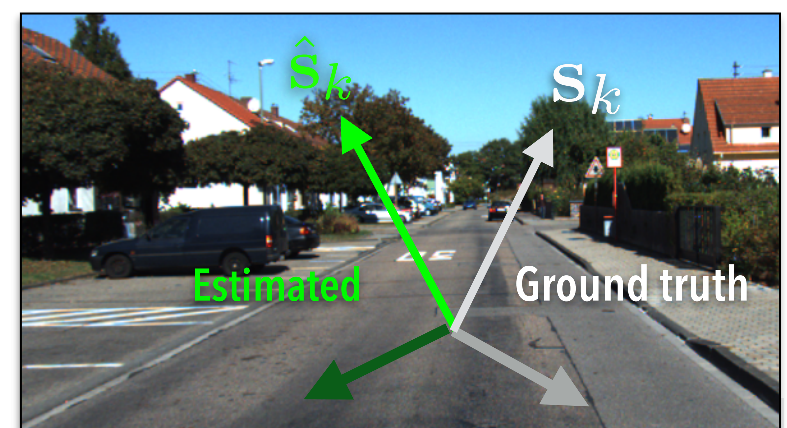
3D sun vector
with uncertainty





Sample a posterior, use Monte Carlo integration
to approximate mean and covariance

$$\mathcal{N}\left(\begin{bmatrix}\bar{\theta}\\ \bar{\phi}\end{bmatrix}, \begin{bmatrix}\sigma_\theta^2 & \sigma_{\theta\phi}\\ \sigma_{\phi\theta} & \sigma_\phi^2\end{bmatrix}\right)$$



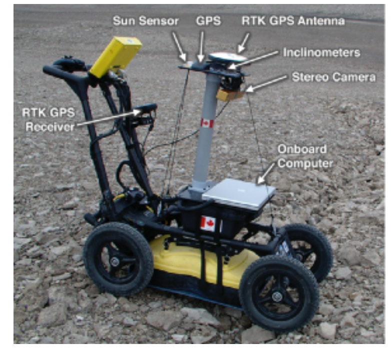
Cosine distance loss

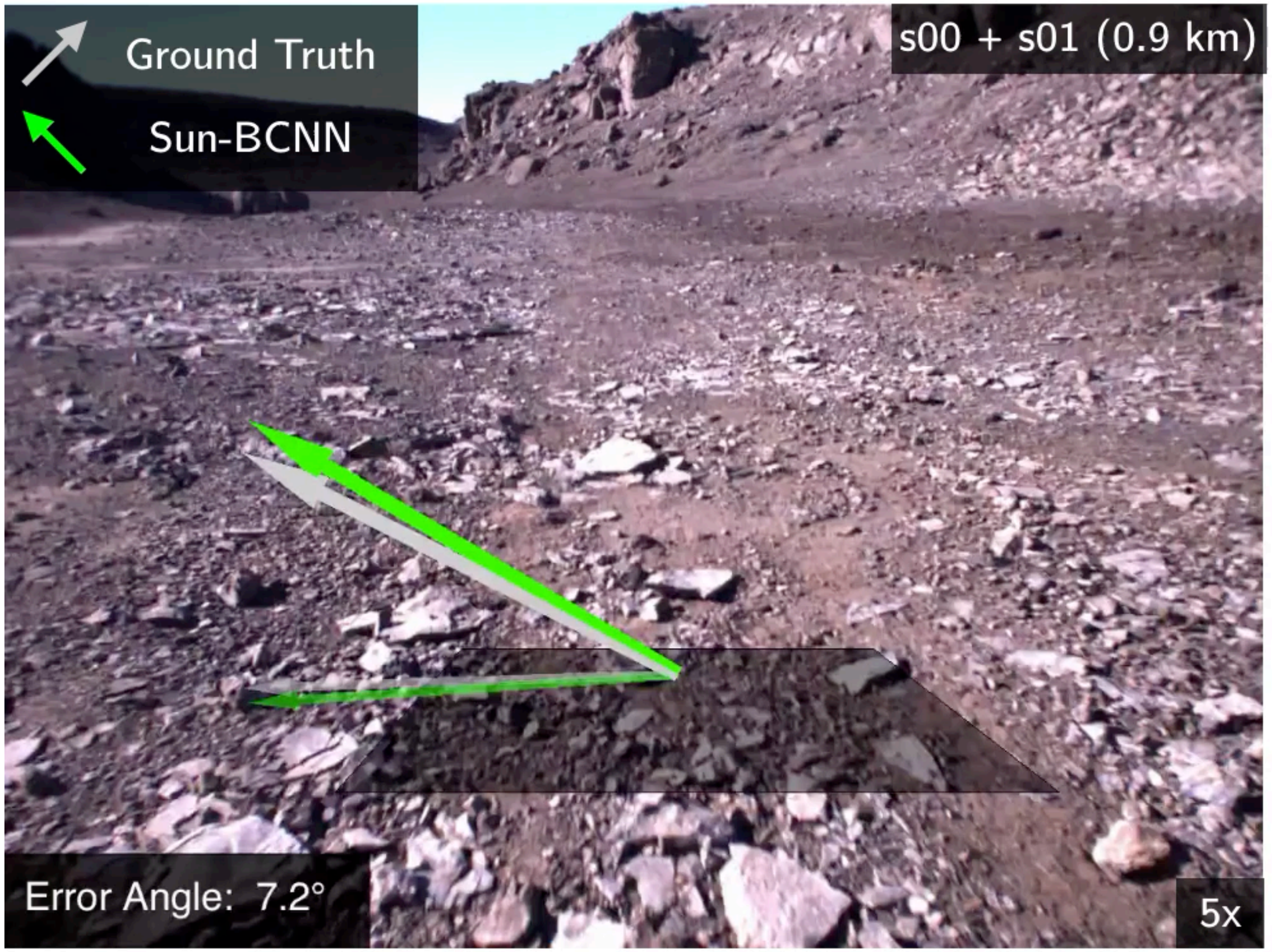
$$\mathcal{L}(\hat{\mathbf{s}}_k, \mathbf{s}_k) = 1 - \hat{\mathbf{s}}_k \cdot \mathbf{s}_k$$

Gal and Ghahramani, "**Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning**" **ICML** (2016)

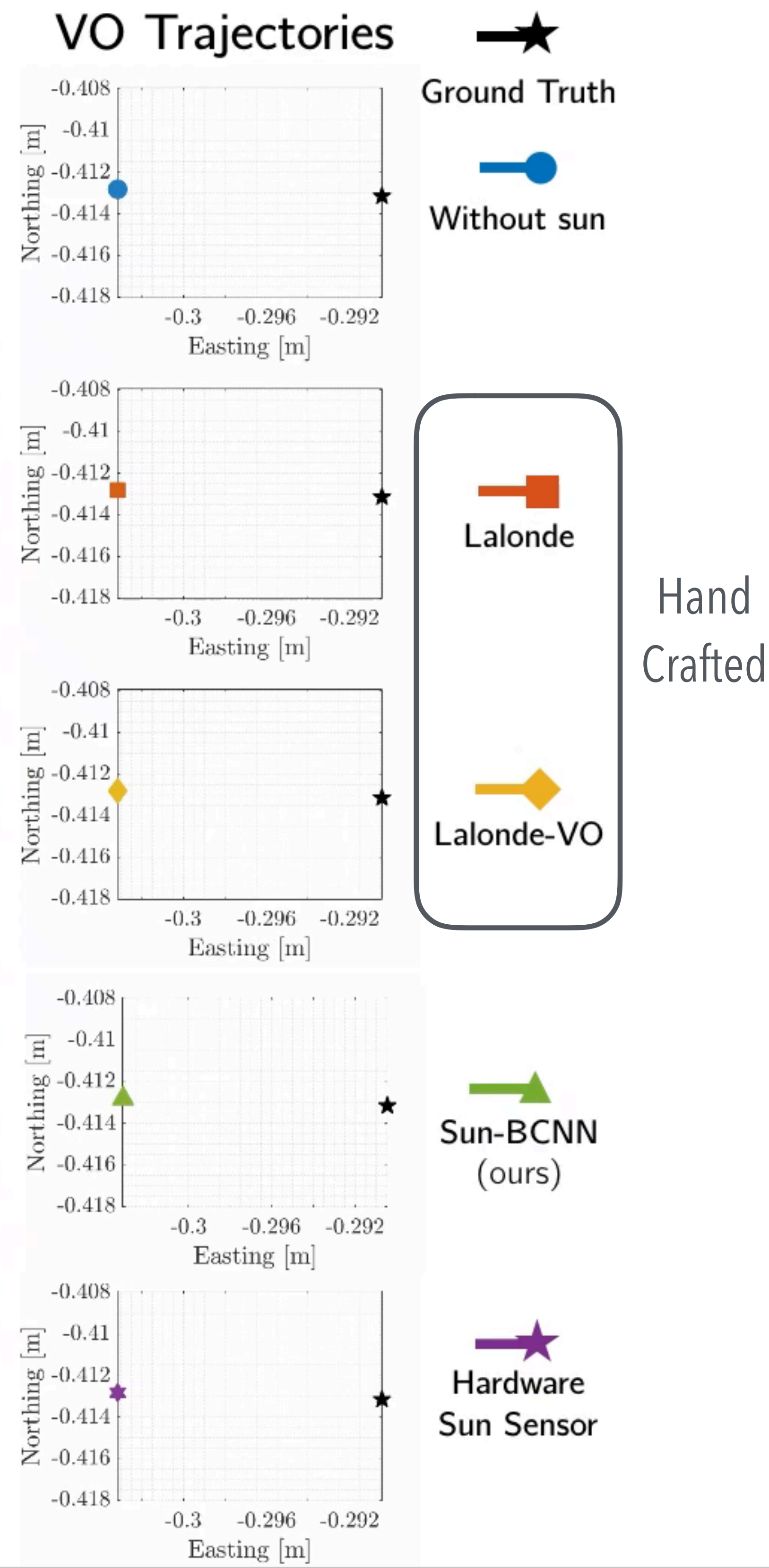
Sun-BCNN Testing

Devon Island



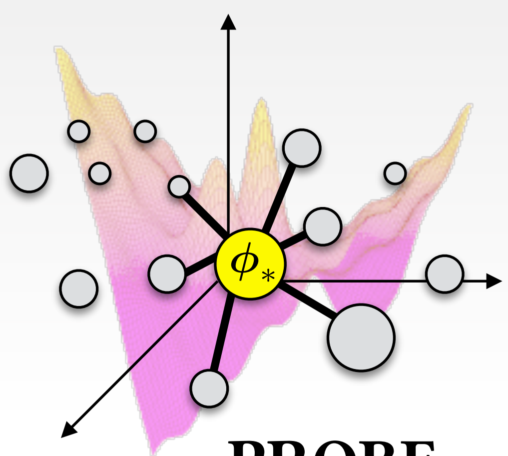


Furgale, **The Devon Island rover navigation dataset**, **IJRR** (2012)



Learned Improvements

I uncertainty quantification

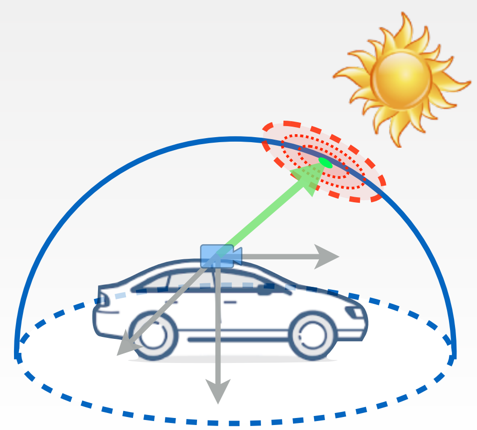


PROBE

Predictive Robust Estimation

IROS 2015
ICRA 2016

II latent representation

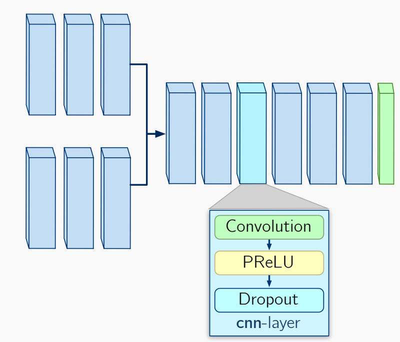


Sun-**BCNN**

Learning Sun Direction with Uncertainty

ISER 2017, ICRA 2017, IJRR 2018

III bias correction

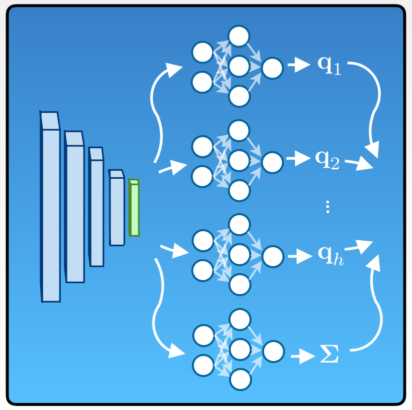


DPC-Net

Learning Estimator Bias through Deep Pose Correction

ICRA / RA-L 2018
ICRA 2020

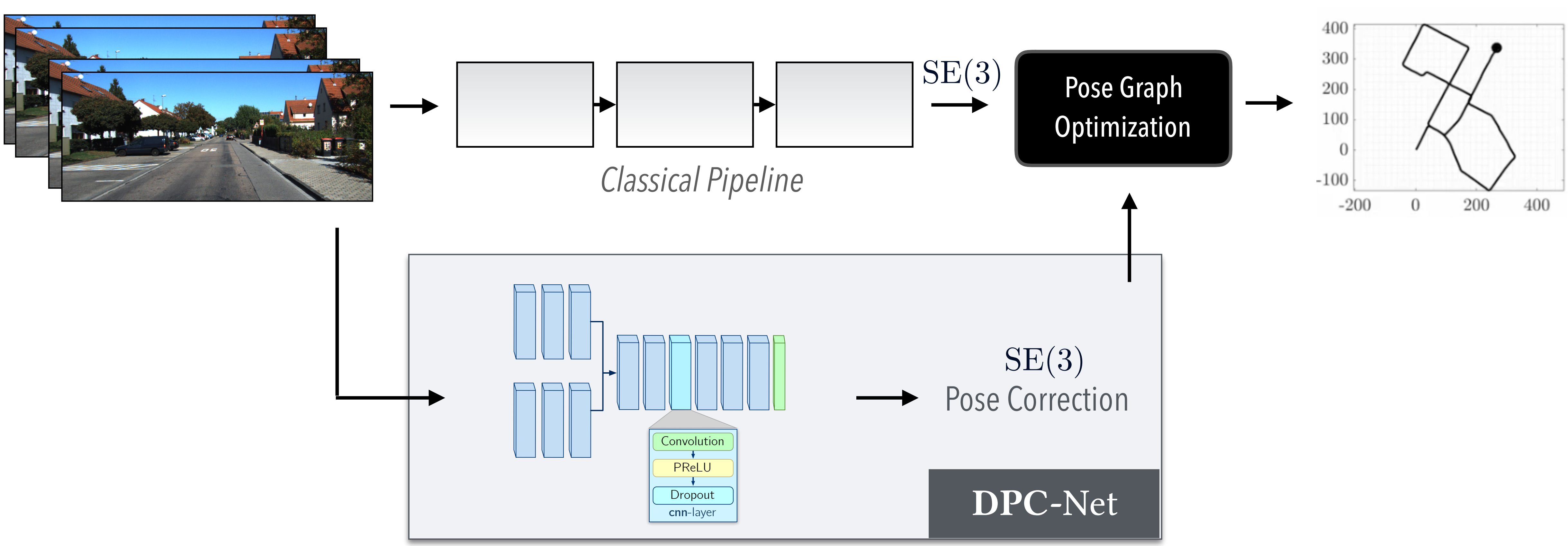
IV latent representation



HydraNet

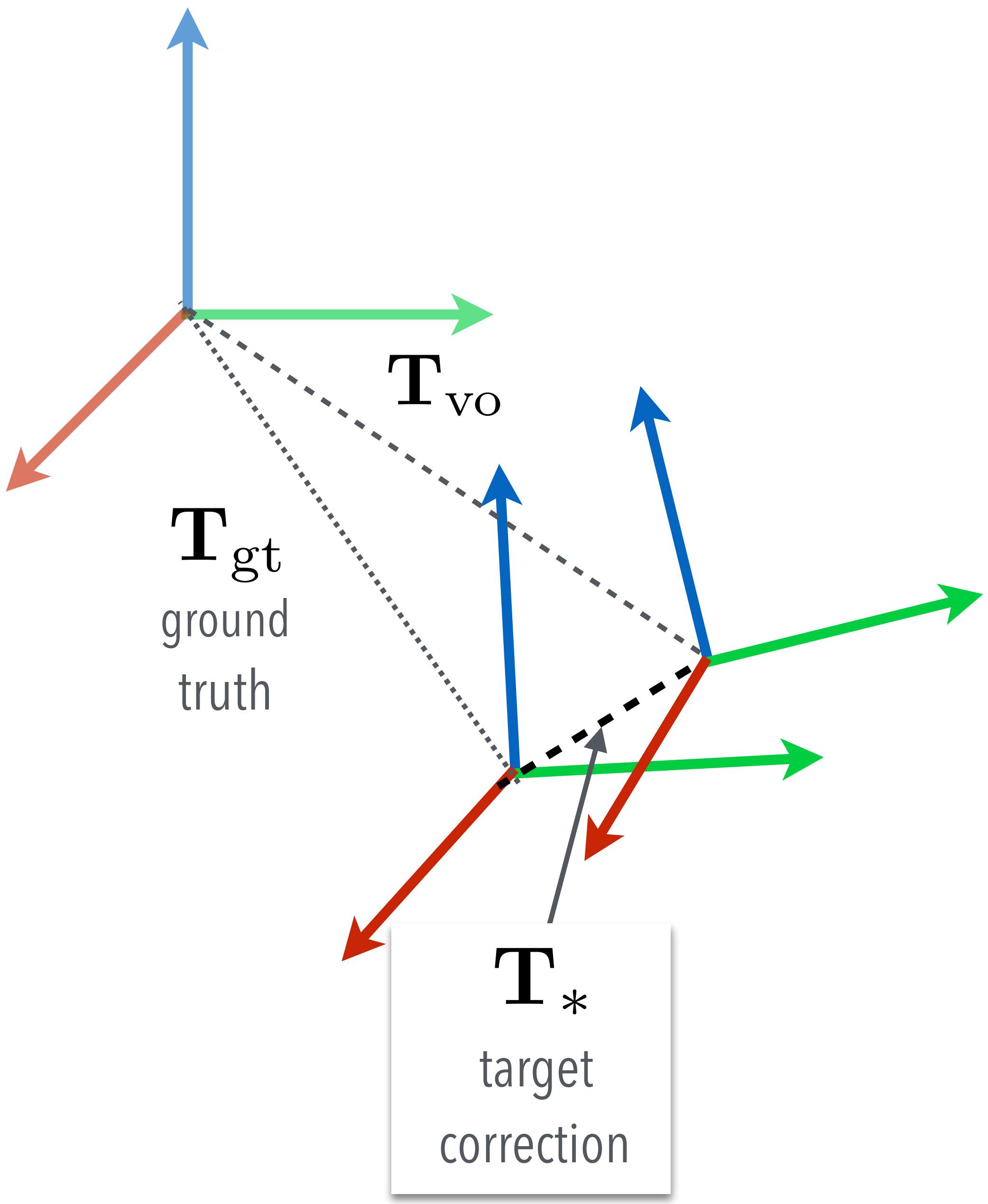
Deep Pose Corrections

Can we generalize Sun-BCNN to learn **SE(3) pose residuals** to correct estimator bias?



Peretroukhin and Kelly, "**DPC-Net: Deep Pose Correction for Visual Localization**," **ICRA / RA-L** (2018)

SE(3) Corrections for Visual Odometry



We learn SE(3) corrections $\mathbf{T}_*$ such that

$$\mathbf{T}_{\mathrm{gt}} = \mathbf{T}_*\mathbf{T}_{\mathrm{vo}}$$

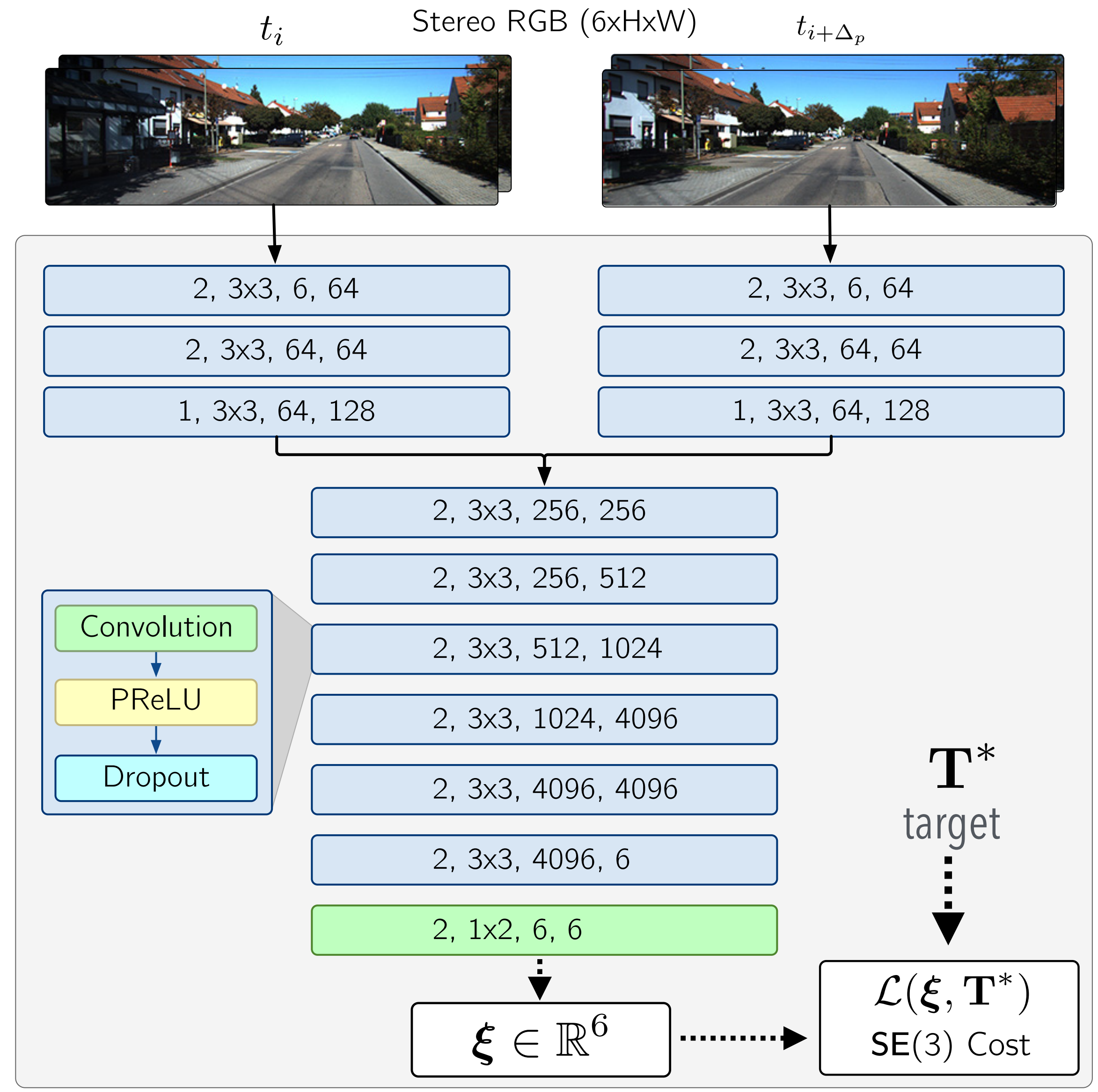
What do we correct for?

- **Estimator biases** (e.g., due to feature distribution or errors in stereo triangulation)
- Intrinsic / extrinsic **mis-calibration**
- **Poor feature tracking** due to blur, localized texture, non-Gaussian residuals

DPC-Net

Structure and Loss

- **DPC-Net** is composed of convolutional layers
- The network takes images as input and outputs an unconstrained vector in the tangent space of identity, $\boldsymbol{\xi}$
- Although the output is unconstrained vector, we store the target corrections, $\mathbf{T}^*$, in matrix form
- We derive a novel loss based on the the **geodesic distance:**

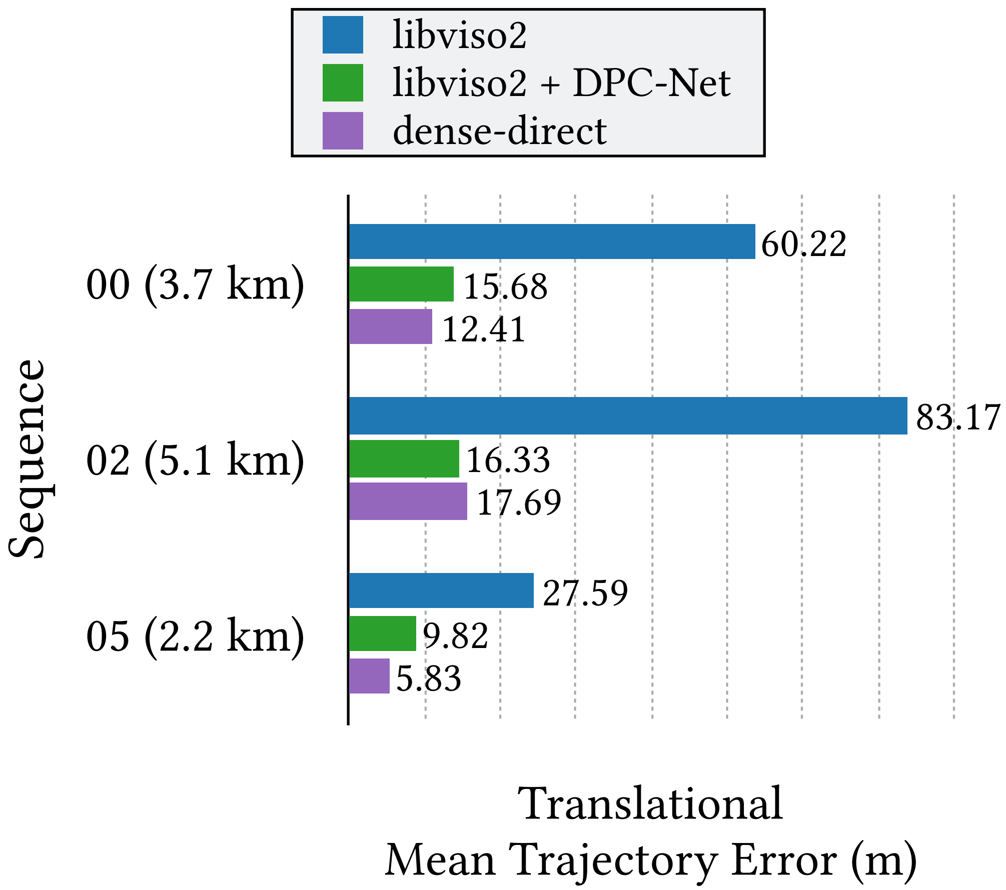


$$\mathcal{L}(\boldsymbol{\xi}) = \frac{1}{2} g(\boldsymbol{\xi})^T \boldsymbol{\Sigma}^{-1} g(\boldsymbol{\xi}) \quad \text{where} \quad g(\boldsymbol{\xi}) \triangleq \log\left(\exp\left(\boldsymbol{\xi}^\wedge\right) \mathbf{T}^{*-1}\right)^\vee$$

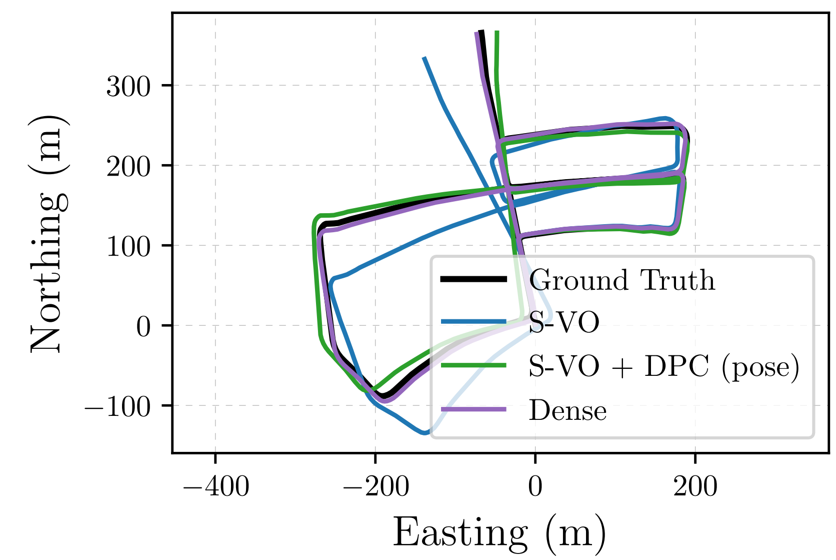
weighs rotation and translation terms
based on training data

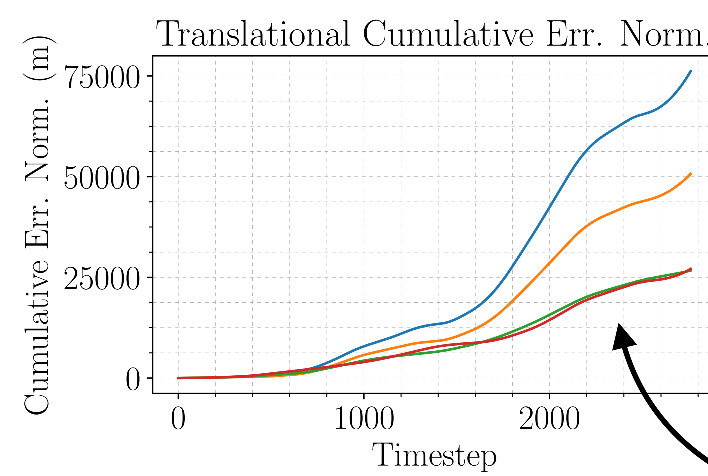
DPC-Net | Correcting `libviso2`

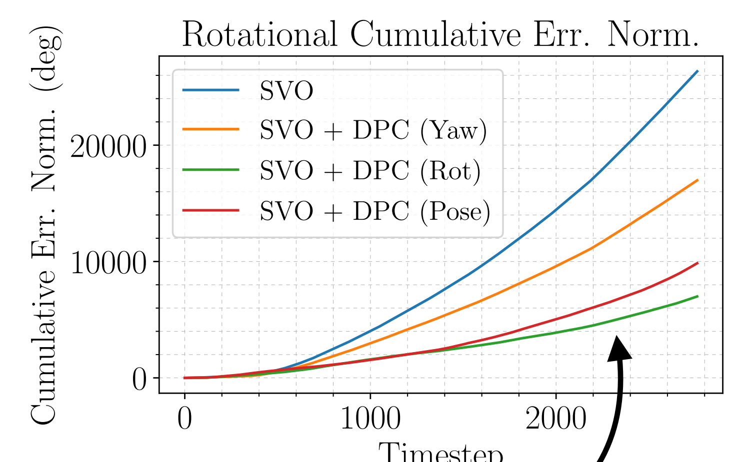
- We evaluated DPC-Net on the KITTI odometry benchmark and train it to correct an efficient VO estimator based on `libviso2`



Sequence **05** (2.2 km)



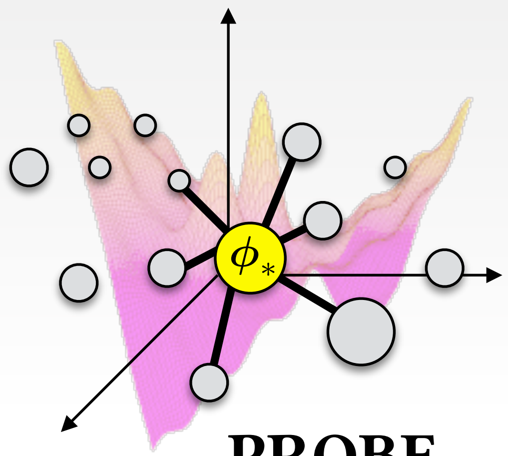




rotation corrections dominate

Learned Improvements

I uncertainty quantification

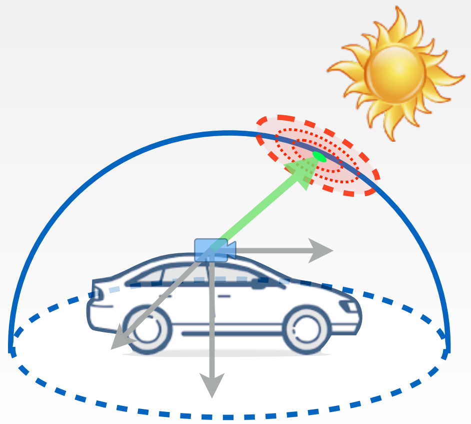


PROBE

Predictive Robust Estimation

IROS 2015
ICRA 2016

II latent representation

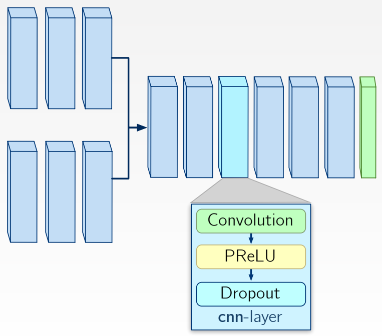


Sun-**BCNN**

Learning Sun Direction with Uncertainty

ISER 2017, ICRA 2017, IJRR 2018

III bias correction

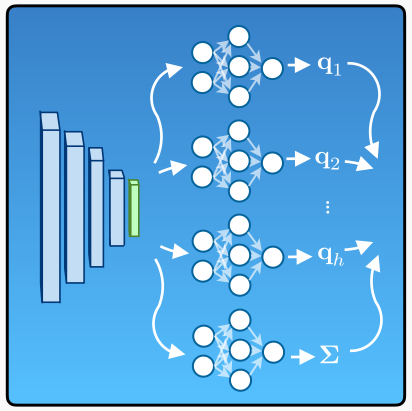


DPC-Net

Learning Estimator Bias through Deep Pose Correction

ICRA / RA-L 2018
ICRA 2020

IV latent representation



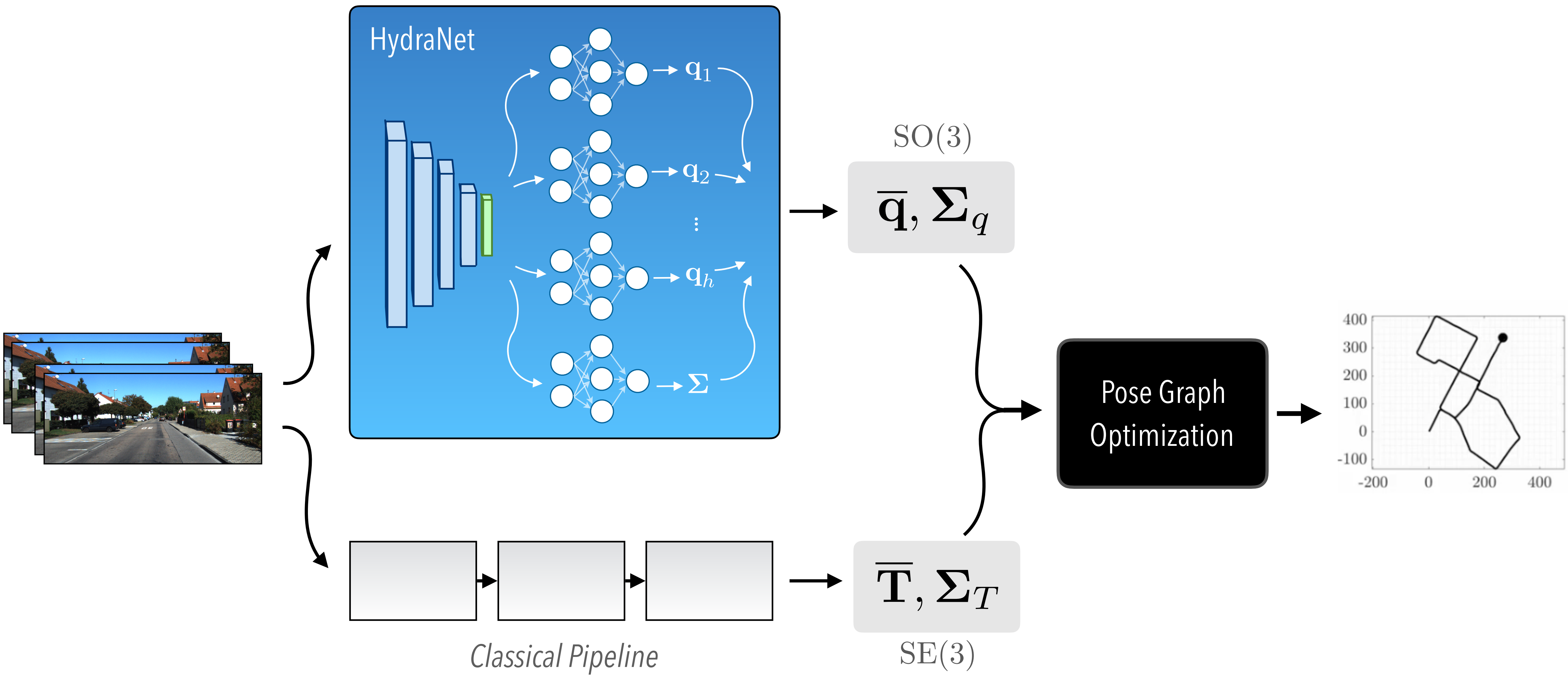
HydraNet

Learning Rotation With Uncertainty

CVPR Workshops 2019

Deep Rotation Regression with Uncertainty

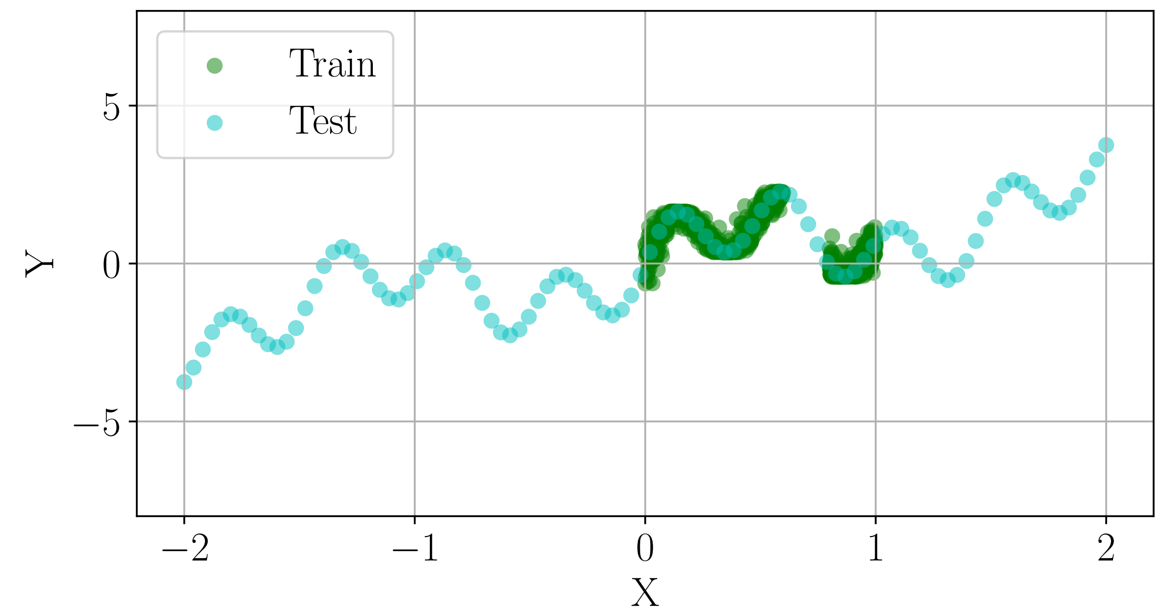
Can learned estimates of camera rotation (with *uncertainty*) improve visual egomotion?

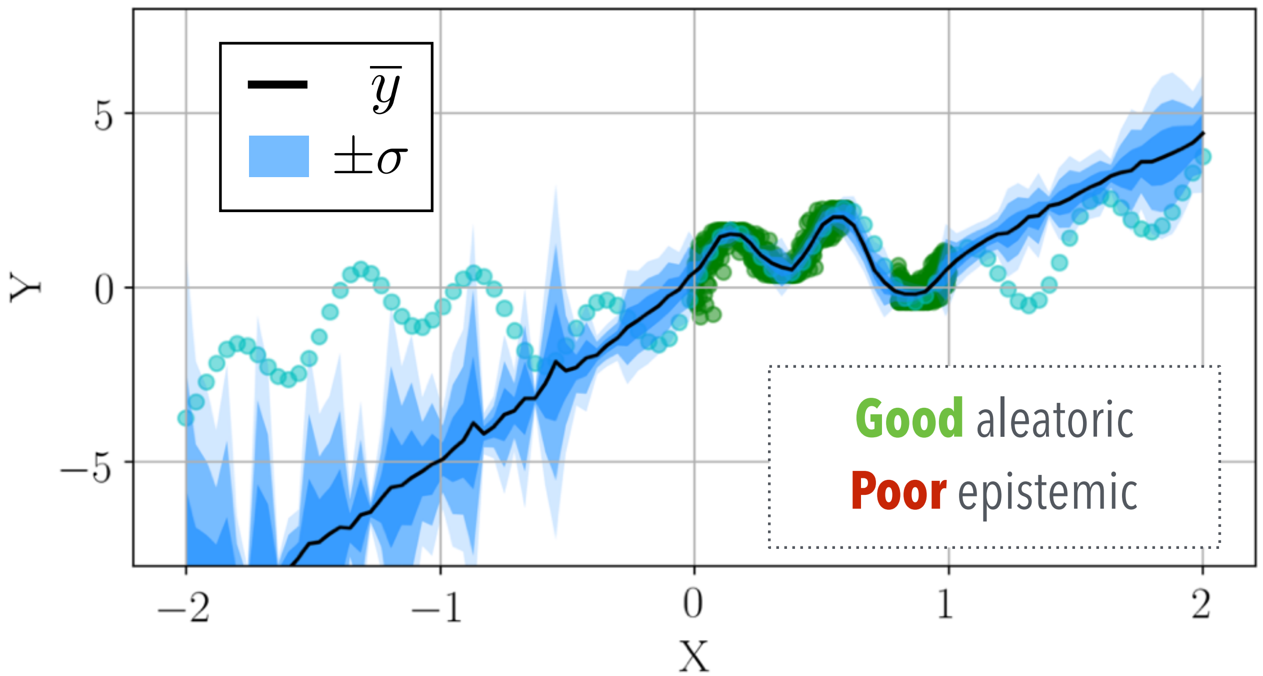


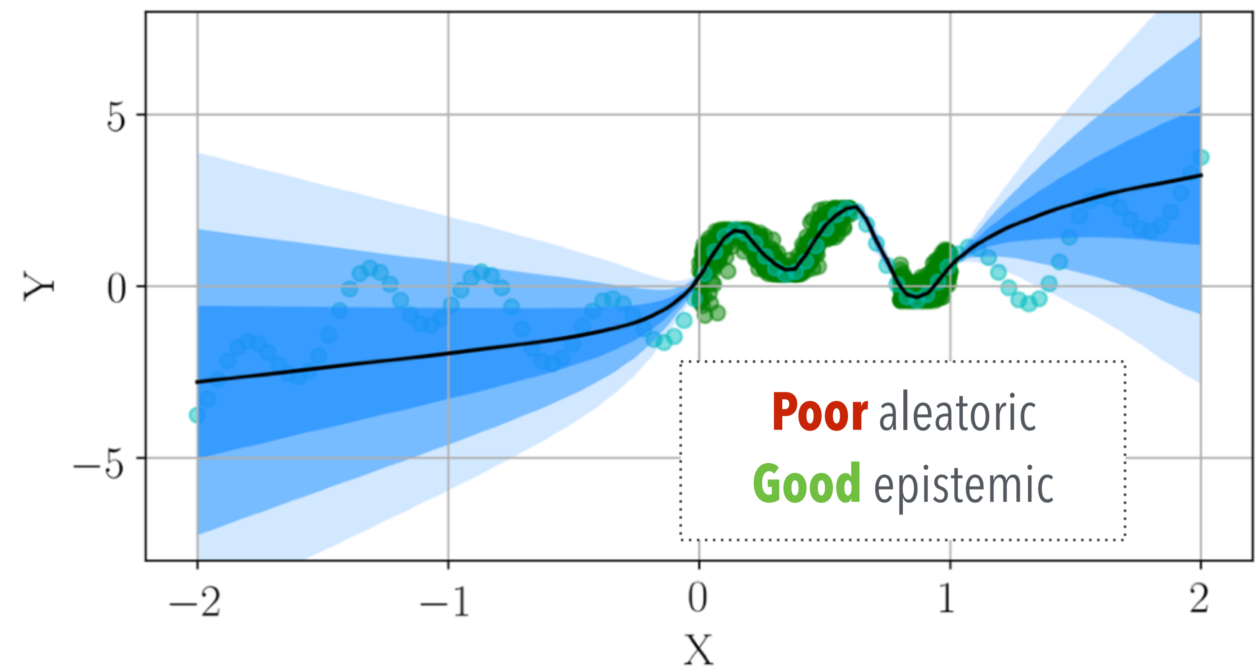
Kneip, Siegwart, and Pollefeys, "**Finding the Exact Rotation between Two Images Independently of the Translation**," **ECCV** (2012)
Peretroukhin, Wagstaff and Kelly, "**Deep Probabilistic Regression of Elements of SO(3)**," **CVPR,** *Workshop on Uncertainty and Robustness in Deep Visual Learning* (2019)

Sources of Uncertainty

1. **Aleatoric** (*'observation' noise*)
   - A result of the underlying process (*e.g., sensor noise*)
2. **Epistemic** (*'model' uncertainty*)
   - A result of a 'distance' between training data and test data



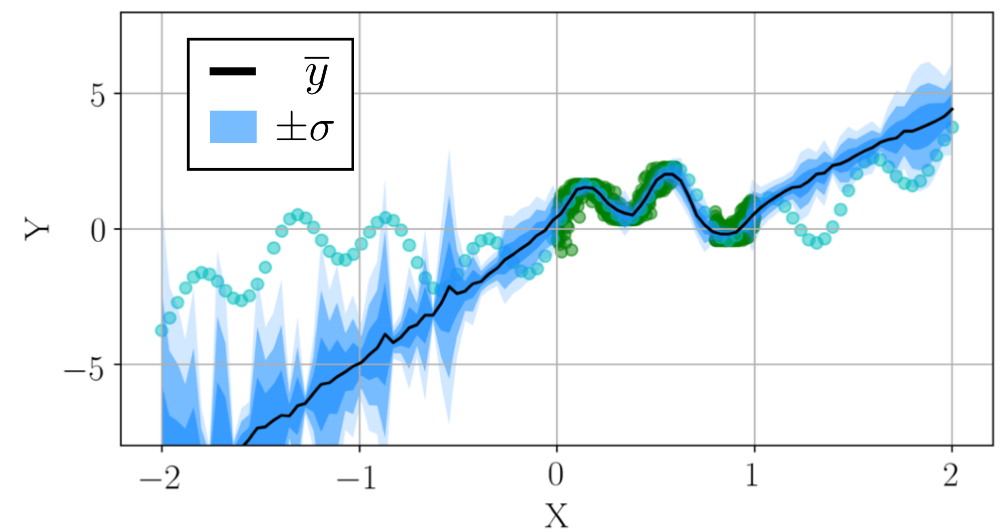




Uncertainty in Neural Networks

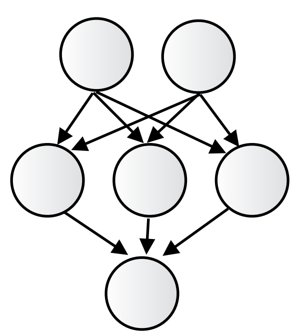
- Monte Carlo dropout (technique used by Sun-BCNN) also often **poorly captures epistemic uncertainty**
- Some proposed alternatives:

MC Dropout Regression



Direct Covariance Learning

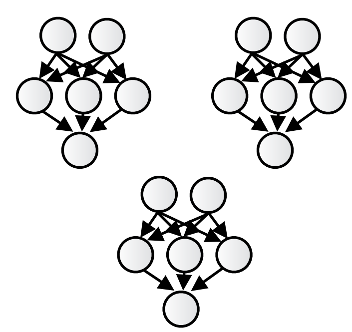
$$\mathcal{L}(y, \sigma^2, y_t) = \frac{1}{\sigma^2}(y - y_t)^2 + \log \sigma^2$$



$$y, \sigma^2$$

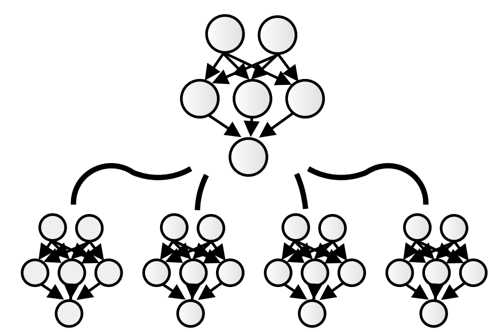
Ensembles

(Bootstrap Aggregating)



$$\overline{y}, \mathrm{var}(\{y_i\})$$

HydraNet



$$\overline{y}, \mathrm{var}(\{y_i\}) + \sigma^2$$

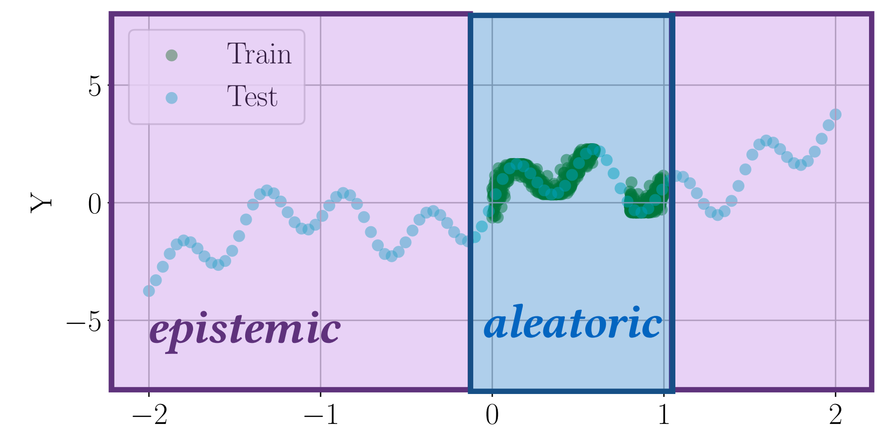
Lernaean Hydra from Greek Mythology

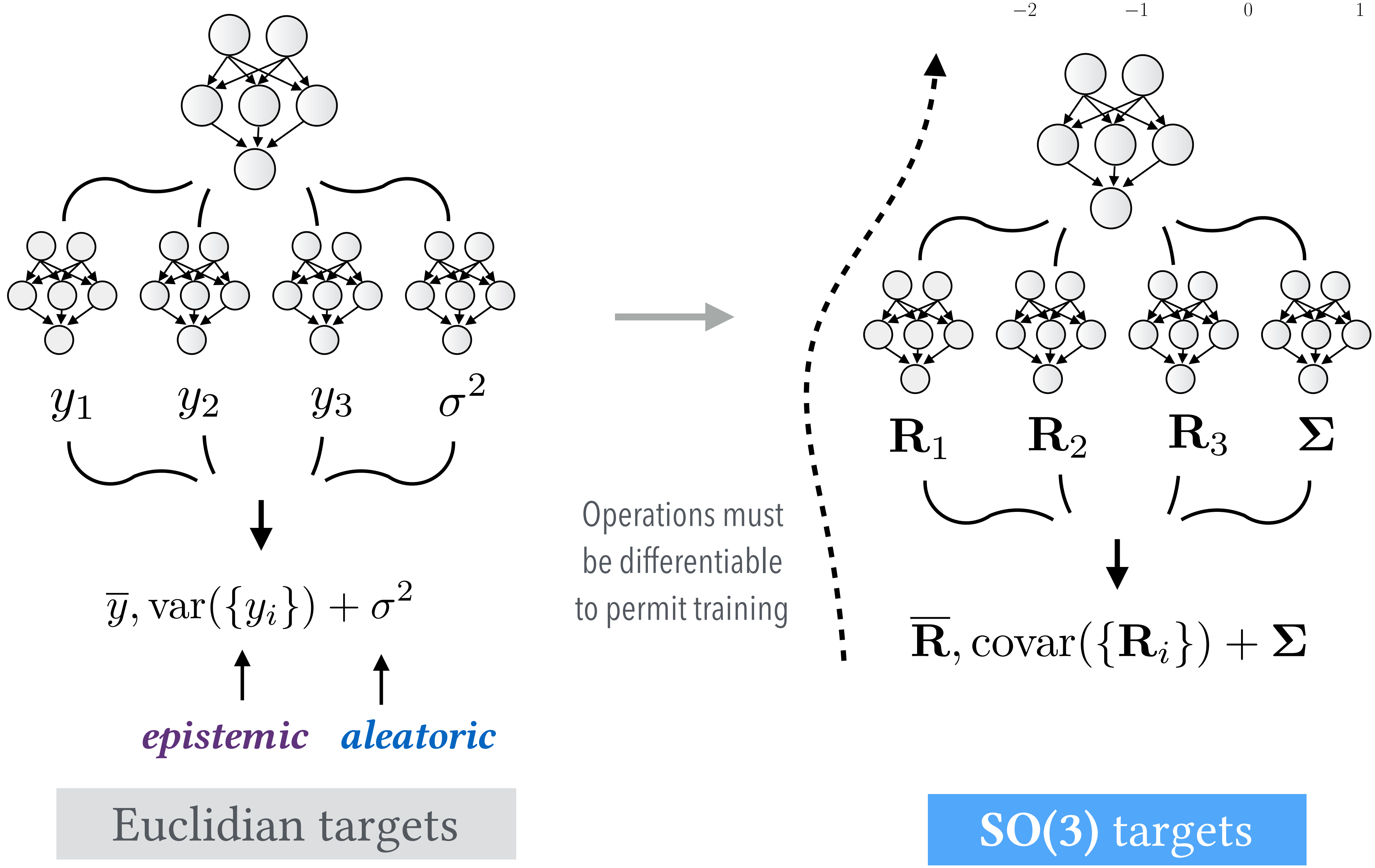
Liu, Ok et al., "**Deep Inference for Covariance Estimation…**", **ICRA** (2018)

Lakshminarayanan et al., "**Simple and scalable predictive uncertainty estimation using deep ensembles,**" **NeurIPS** (2017)

HydraNet

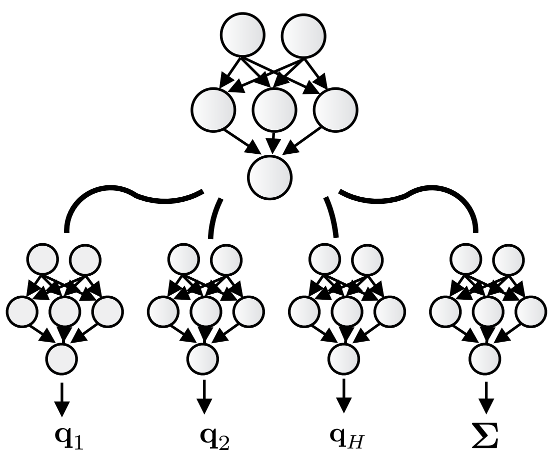
Epistemic and Aleatoric Uncertainty

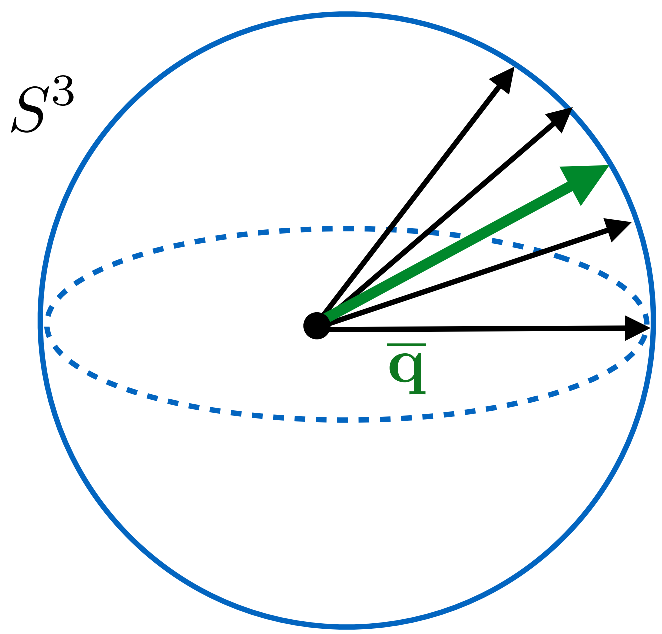




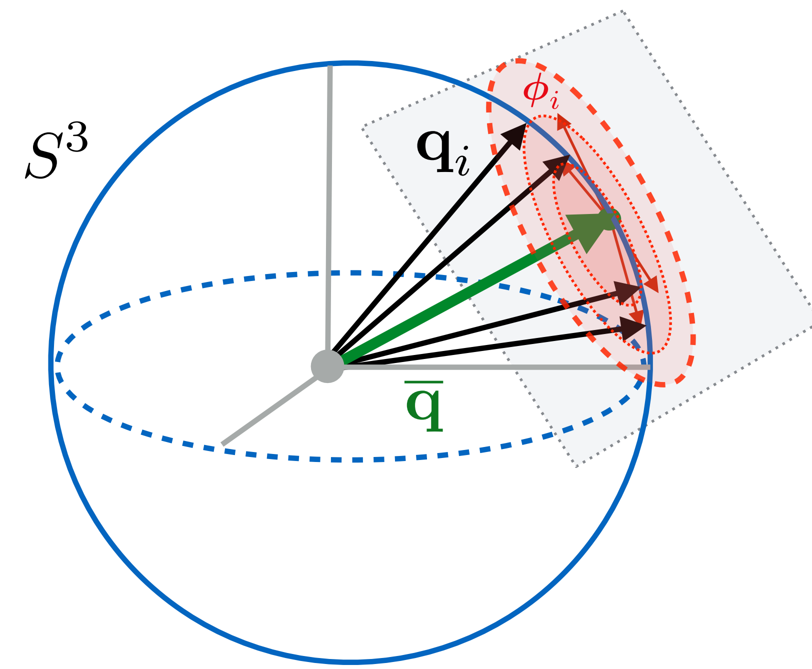
Rotation Averaging

$$\overline{\mathbf{R}} = \underset{\mathbf{R} \in SO(3)}{\operatorname{argmin}} \sum_{i=1}^{n} d(\mathbf{R}_i, \mathbf{R})^2 \longrightarrow \overline{\mathbf{q}} = \underset{\mathbf{q} \in \mathbb{S}^3}{\operatorname{argmin}} \sum_{i=1}^{n} d_{\text{quat}}(\mathbf{q}_i, \mathbf{q})^2$$





$$\overline{\mathbf{q}} = \frac{\sum_{i=1}^{H} \mathbf{q}_i}{\left\| \sum_{i=1}^{H} \mathbf{q}_i \right\|}$$

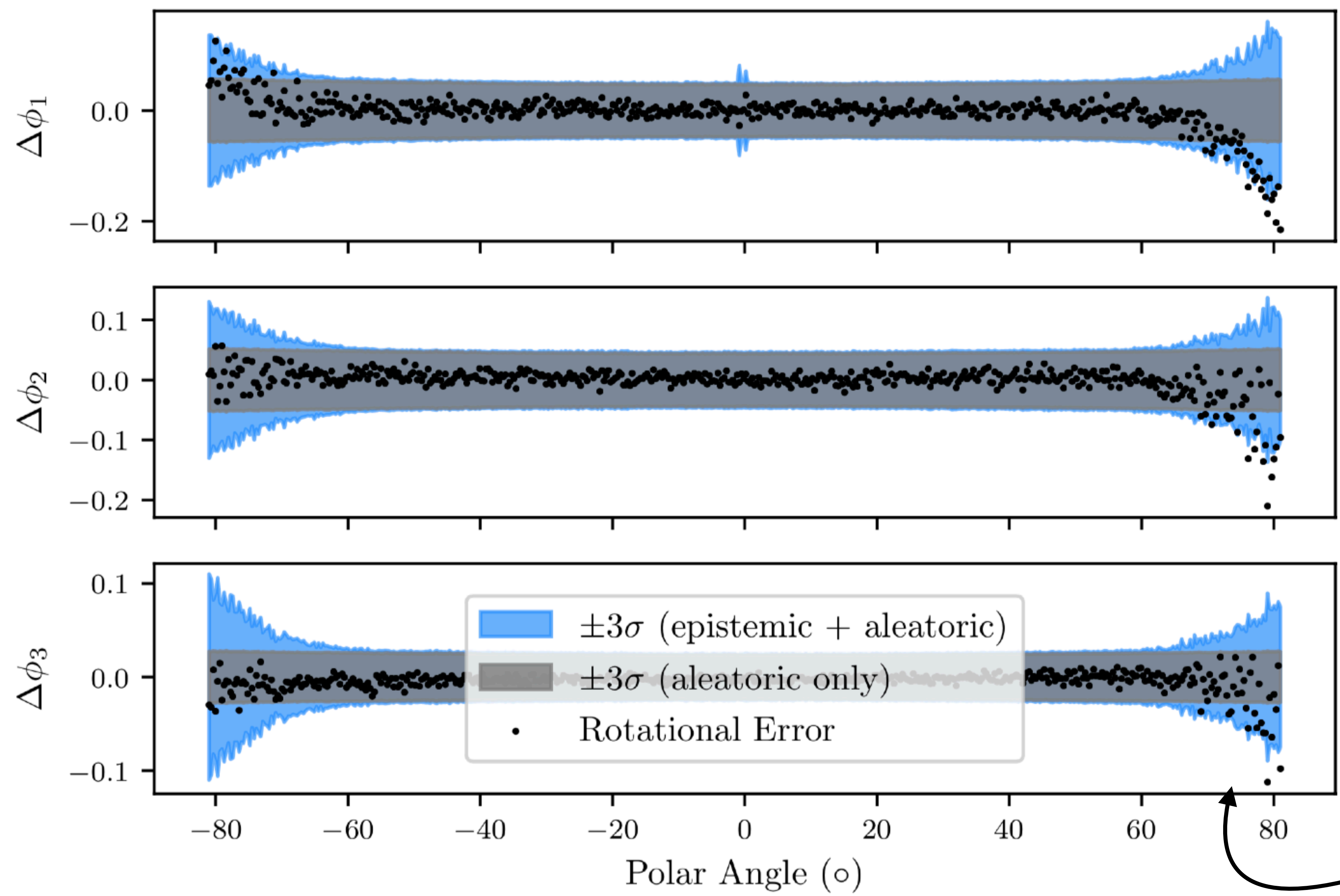


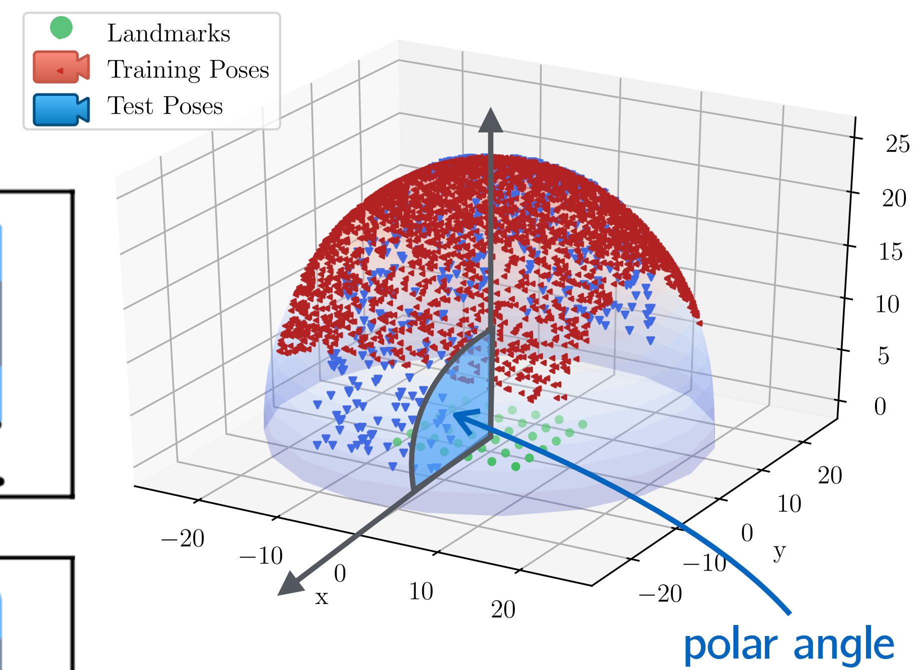
$$\text{covar}(\{\mathbf{q}_i\}) = \frac{1}{H-1} \sum_{i=1}^{H} \phi_i \phi_i^{\mathsf{T}}$$

Hartley et al., "**Rotation Averaging**", **IJCV** (2013)
Sola et al., "**A micro Lie theory for State Estimation in Robotics**", **arXiv** (2019)

Importance of Epistemic Uncertainty

Synthetic ‘orbital’ dataset

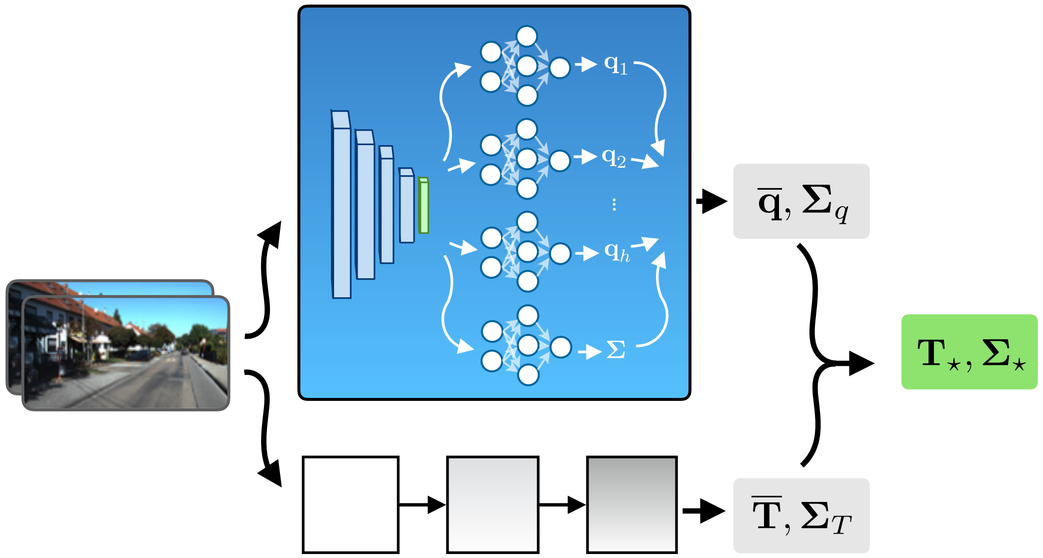


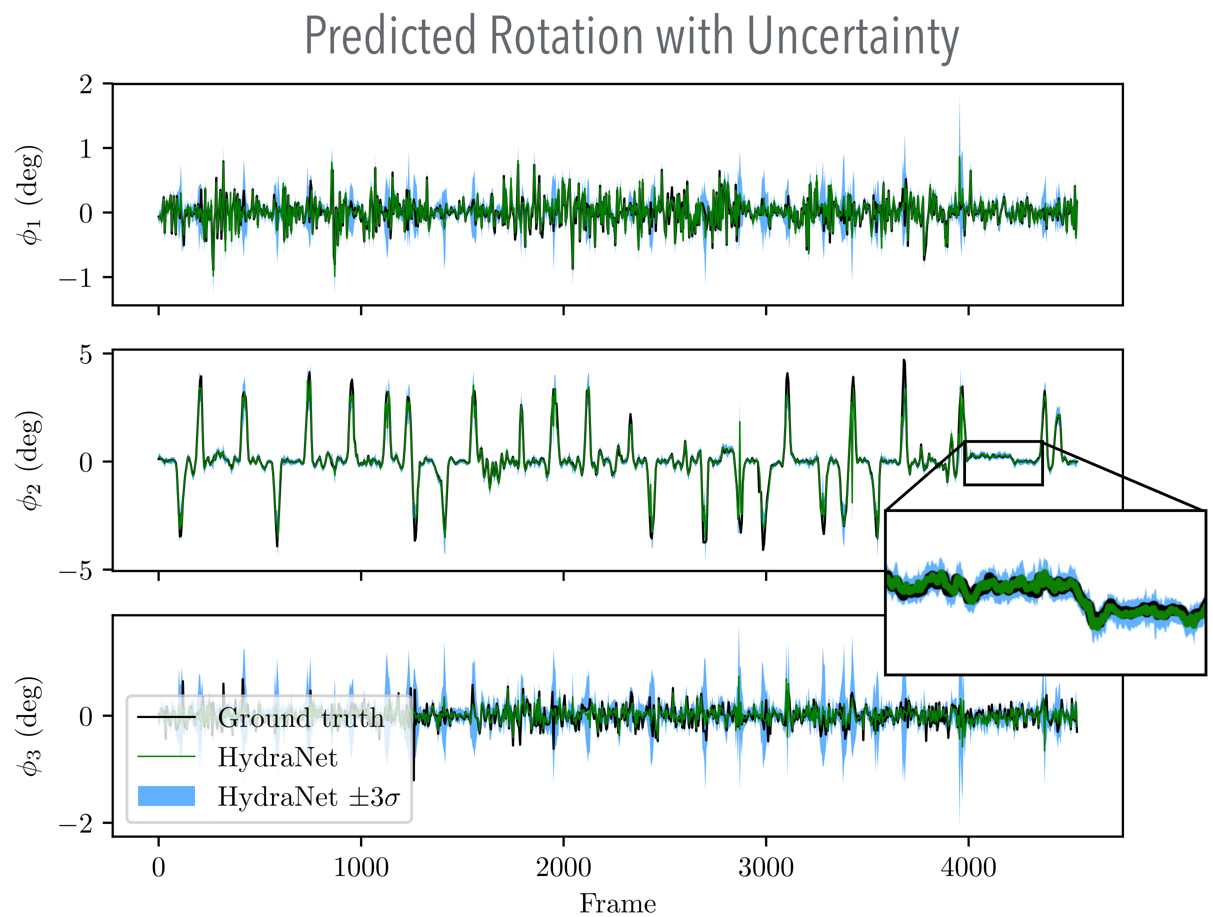


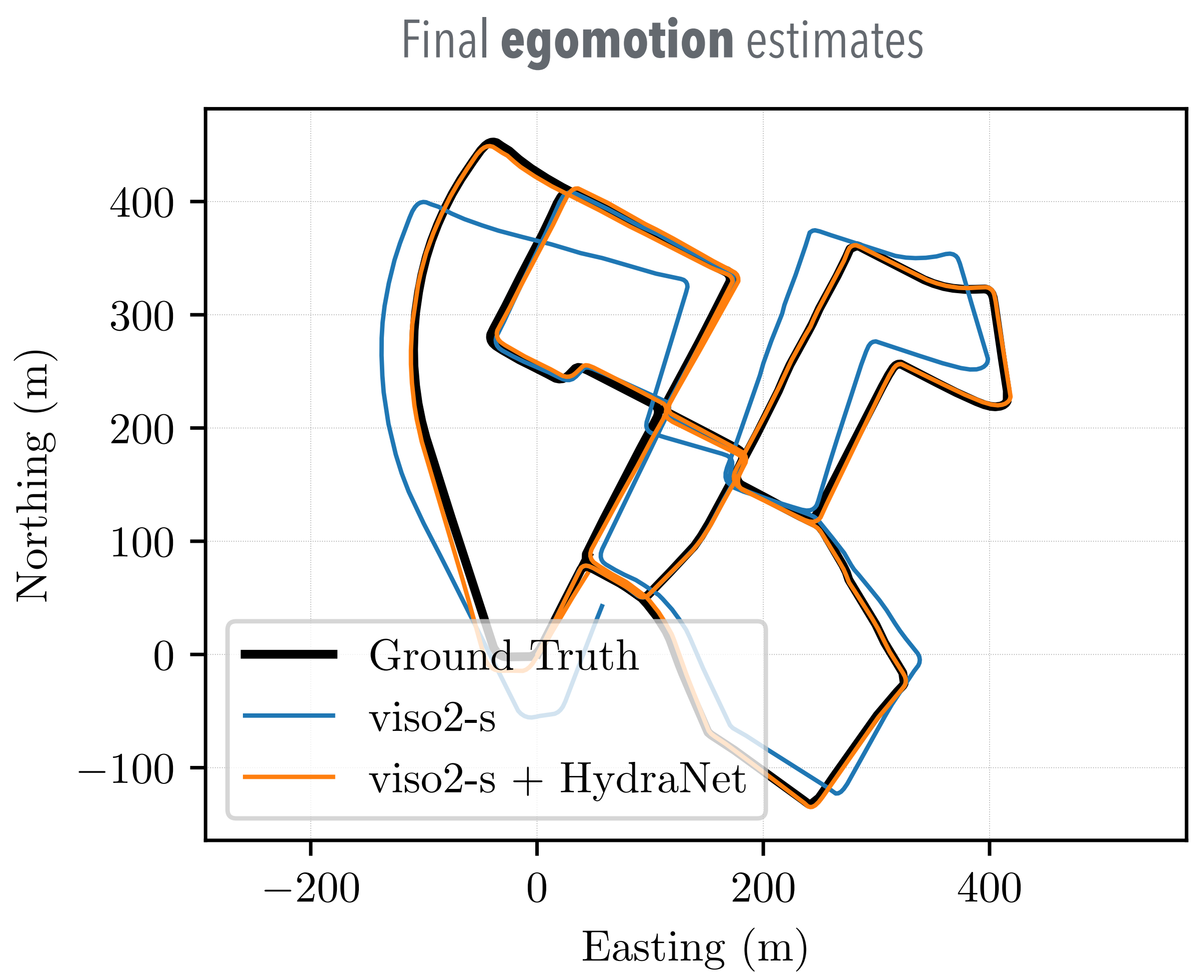
Epistemic uncertainty is necessary to account for ‘out-of-training-distribution’ errors

Sliding Window VO

libviso2 SE(3) *(with uncertainty)* +
HydraNet SO(3) *(with uncertainty)*

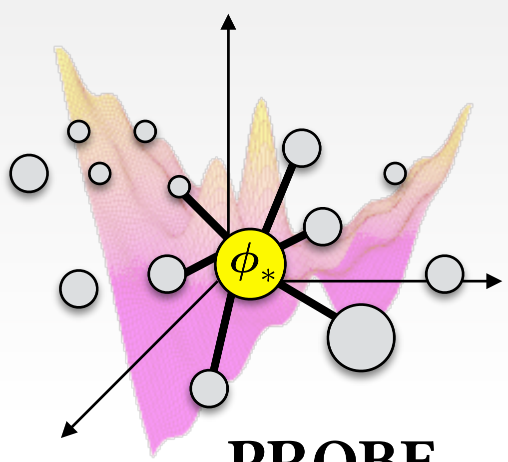






Learned Improvements

I uncertainty quantification



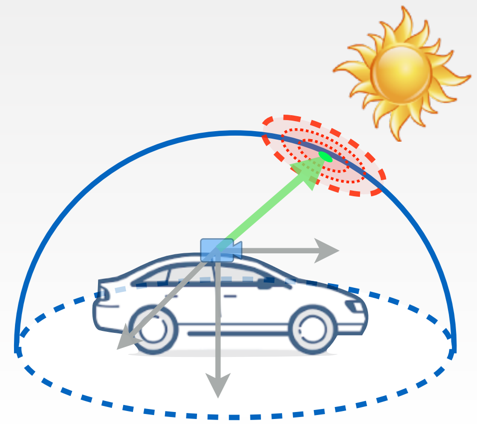
PROBE

Predictive Robust Estimation

IROS 2015

ICRA 2016

II latent representation

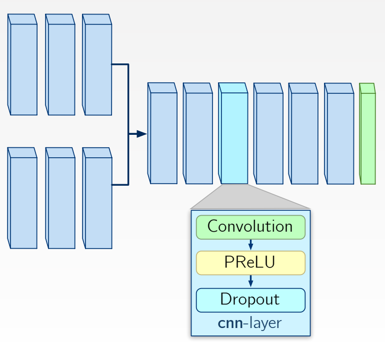


Sun-**BCNN**

Learning Sun Direction with Uncertainty

ISER 2017, ICRA 2017, IJRR 2018

III bias correction



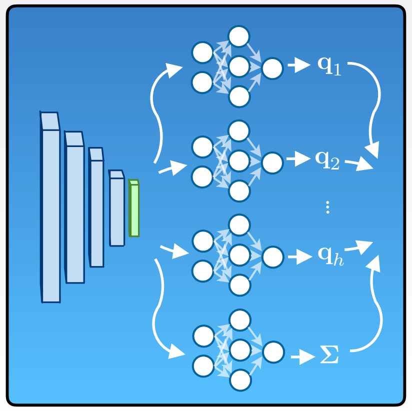
DPC-Net

Learning Estimator Bias through Deep Pose Correction

ICRA / RA-L 2018

ICRA 2020

IV latent representation



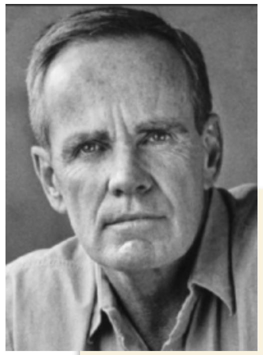
HydraNet

Learning Rotation With Uncertainty

CVPR Workshops 2019

Concluding Remarks

On the Synthesis of Learning and Classical Modelling



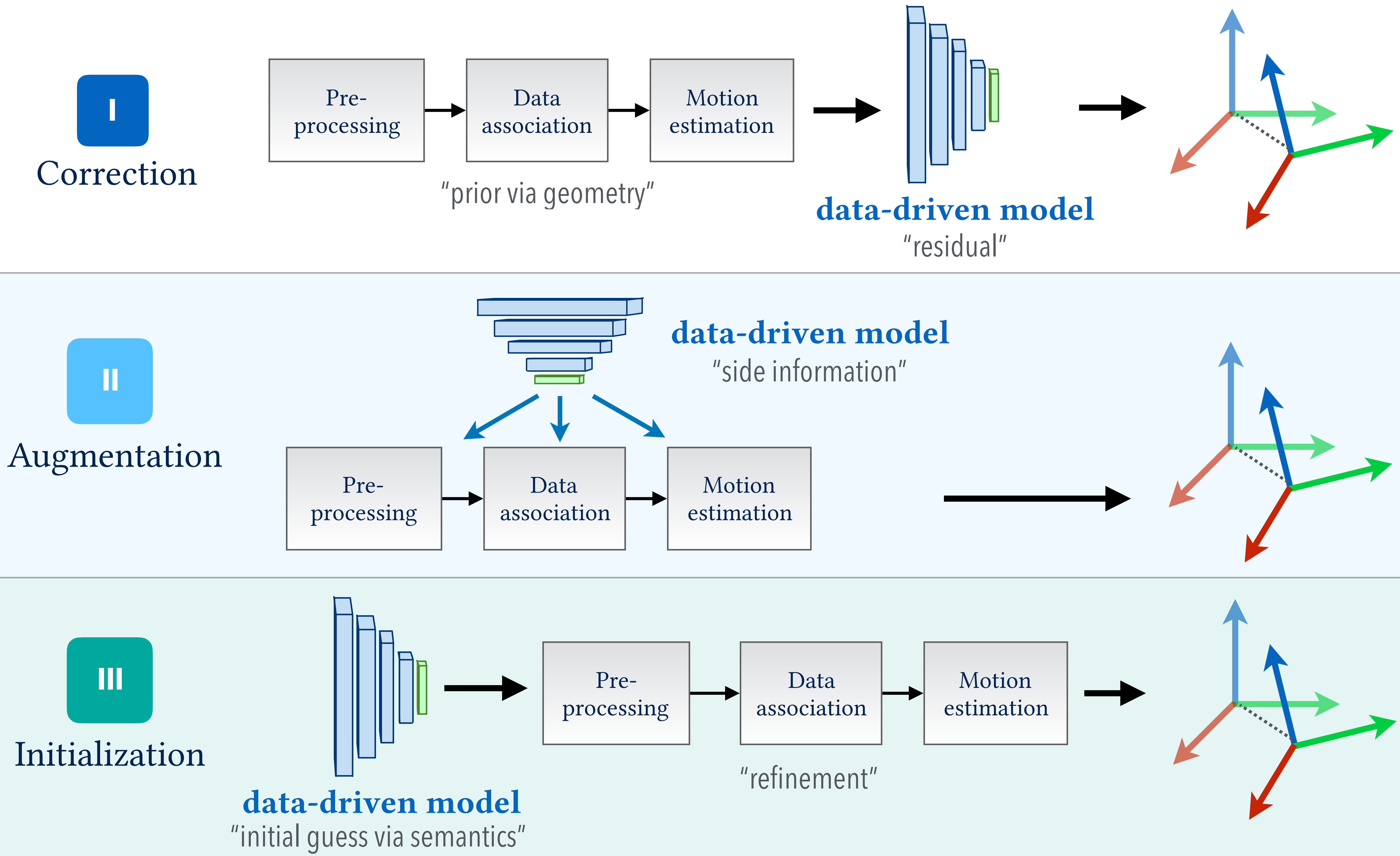
...in this world more things exist without our knowledge than with it and the order in creation which you see is that which you have put there, like a string in a maze...

Cormac McCarthy, Blood Meridian, or the Evening Redness in the West (1985)

The miracle of the appropriateness of the language of mathematics for the formulation of the laws of physics is a wonderful gift which we neither understand nor deserve.

E.P. Wigner, The Unreasonable Effectiveness of Mathematics in the Natural Sciences (1960)

Three Methods of Synthesis



A heartfelt *thank you* to...

my **lab mates**

my **advisors**

Jonathan
Kelly

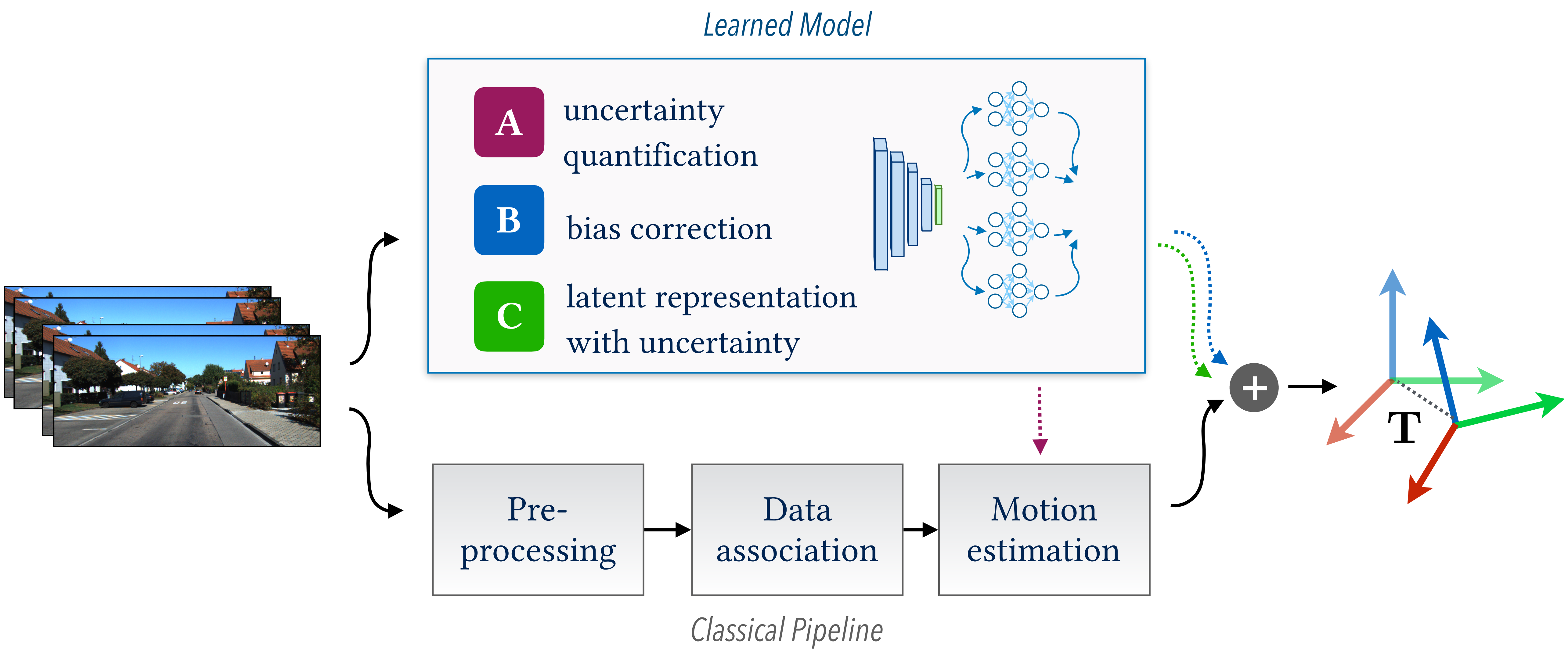
Angela
Schoellig

Tim
Barfoot

and my **friends** and **family** 🤗

Learned Improvements to the Visual Egomotion Pipeline

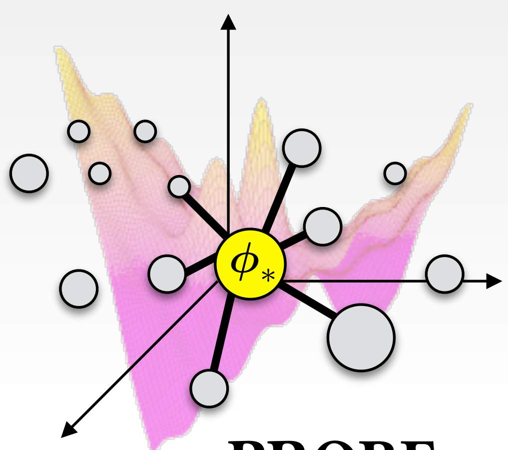
On the Synthesis of Learning and Classical Modelling



Additional Slides

Learned Improvements

I uncertainty quantification



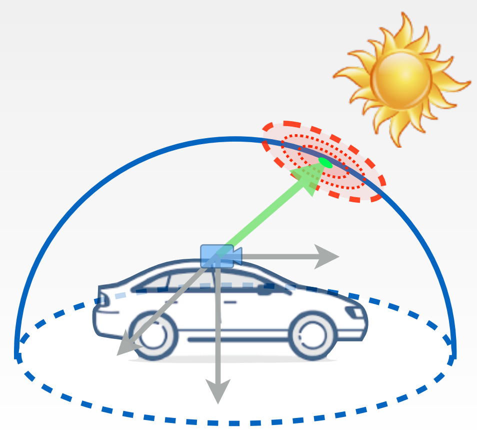
PROBE

Predictive Robust Estimation

IROS 2015

ICRA 2016

II latent representation

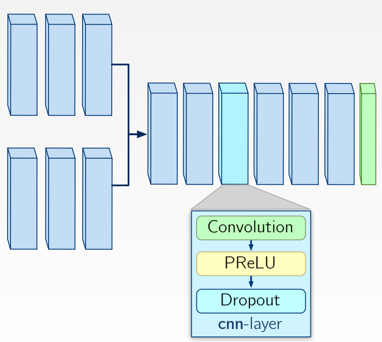


Sun-**BCNN**

Learning Sun Direction with Uncertainty

ISER 2017, ICRA 2017, IJRR 2018

III bias correction



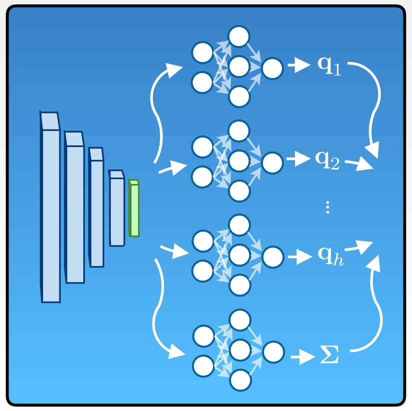
DPC-Net

Learning Estimator Bias through Deep Pose Correction

ICRA / RA-L 2018

ICRA 2020

IV latent representation



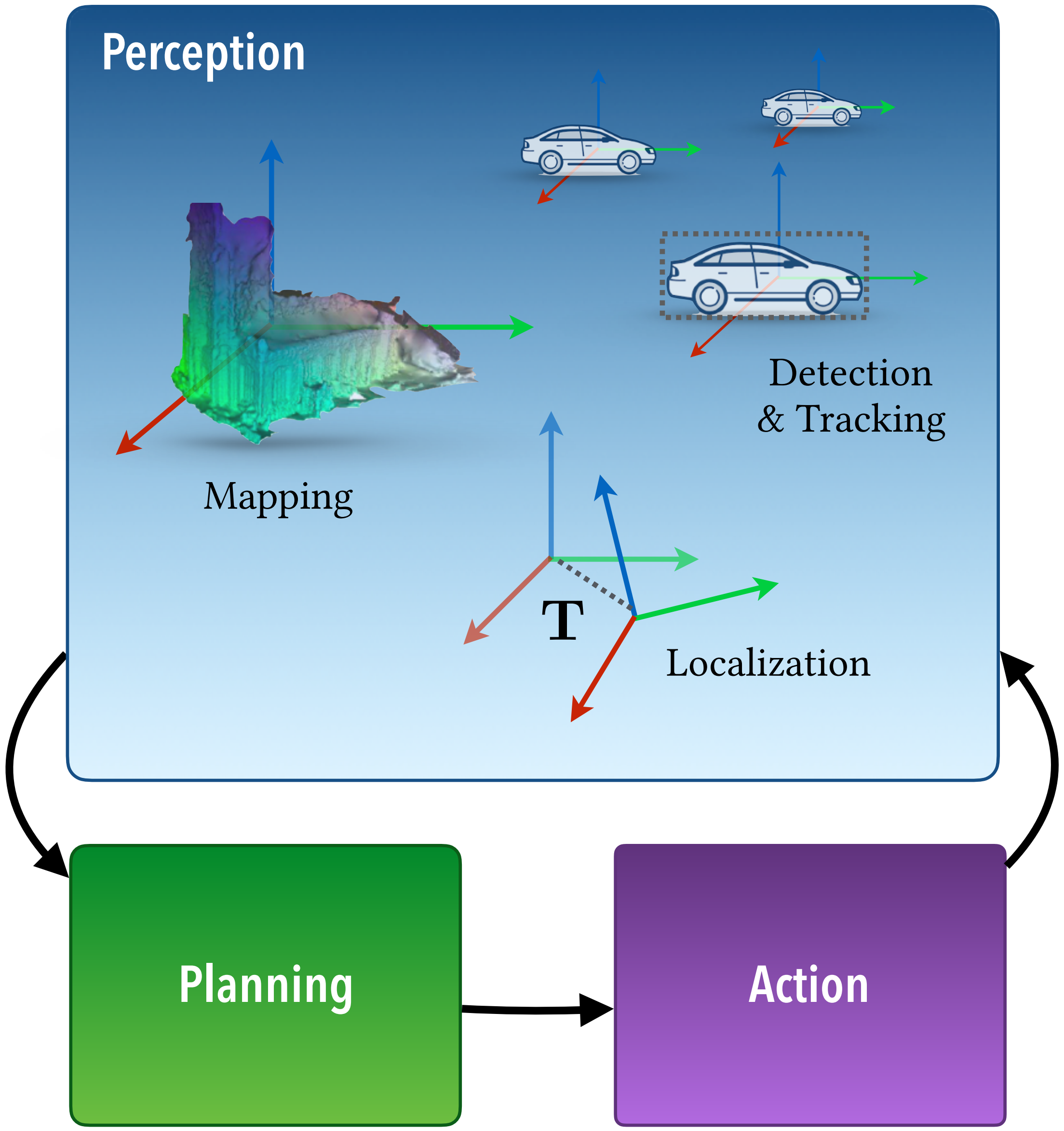
HydraNet

Learning Rotation With Uncertainty

CVPR Workshops 2019

Visual Egomotion

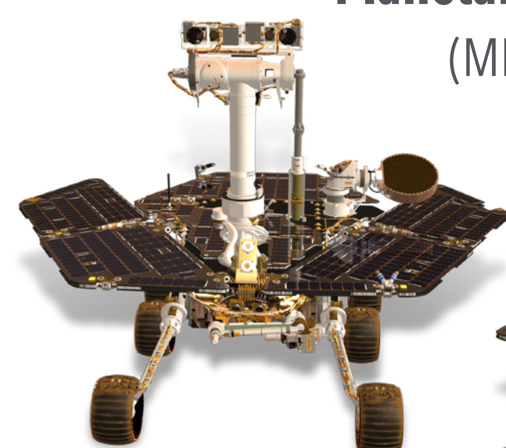
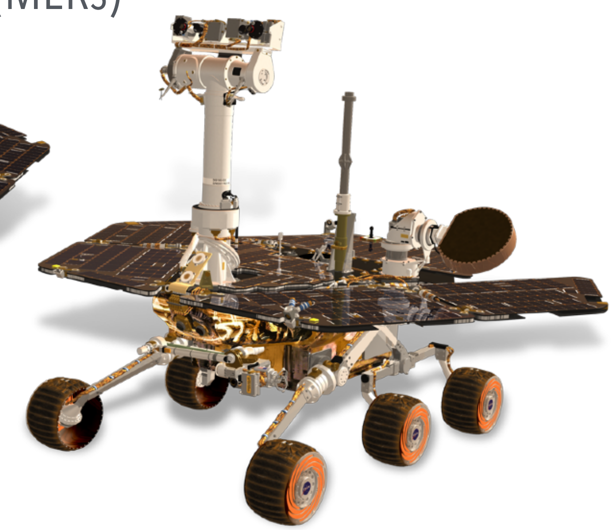
A Core Component of Vision-based Autonomy



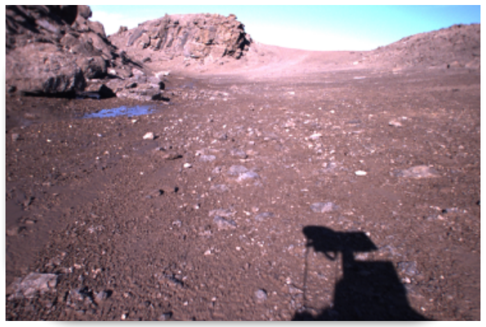
Aerial autonomy
(Skydio 2 Drone)

Assistive Driving
(Tesla Roadster)

Planetary Rovers
(MERs)

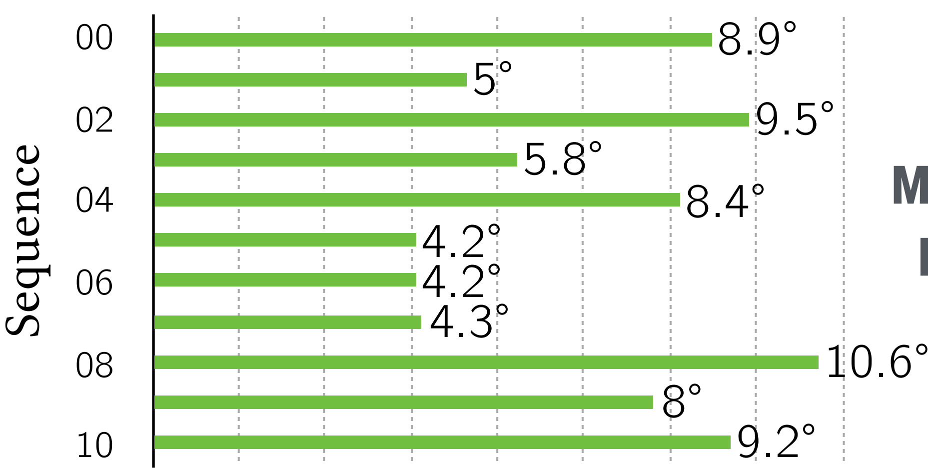



Sun-BCNN Testing
Devon Island and KITTI

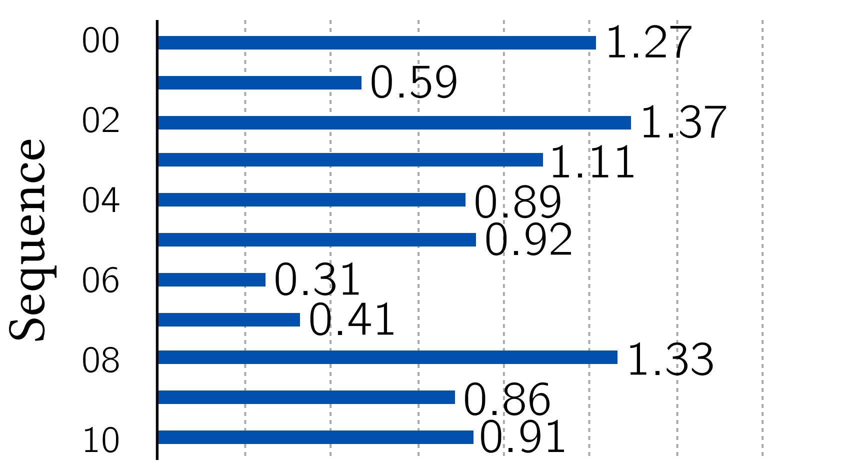


Ground truth
Sinclair Interplanetary digital sun sensor

Ground truth
6-DoF GPS-INS and a solar ephemeris model (based on GPS timestamp)

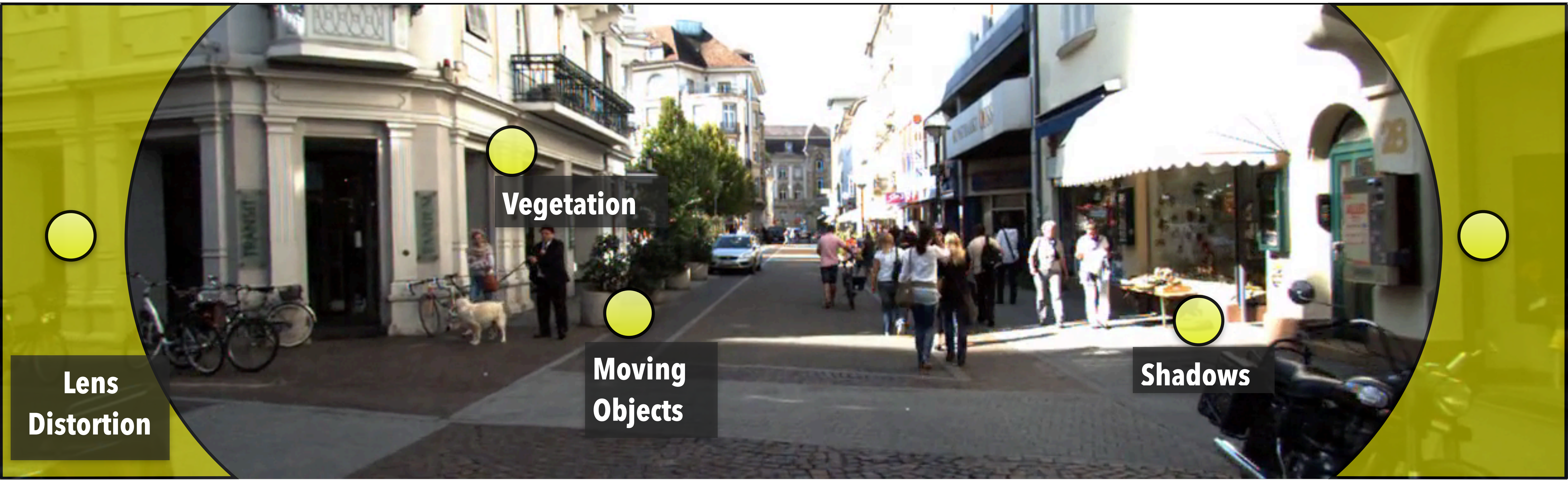


Median Error

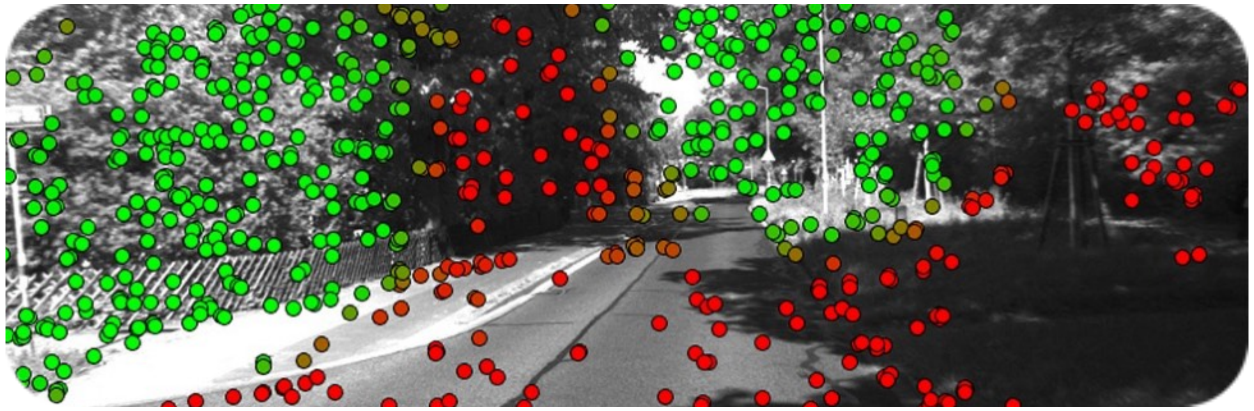


ANEES

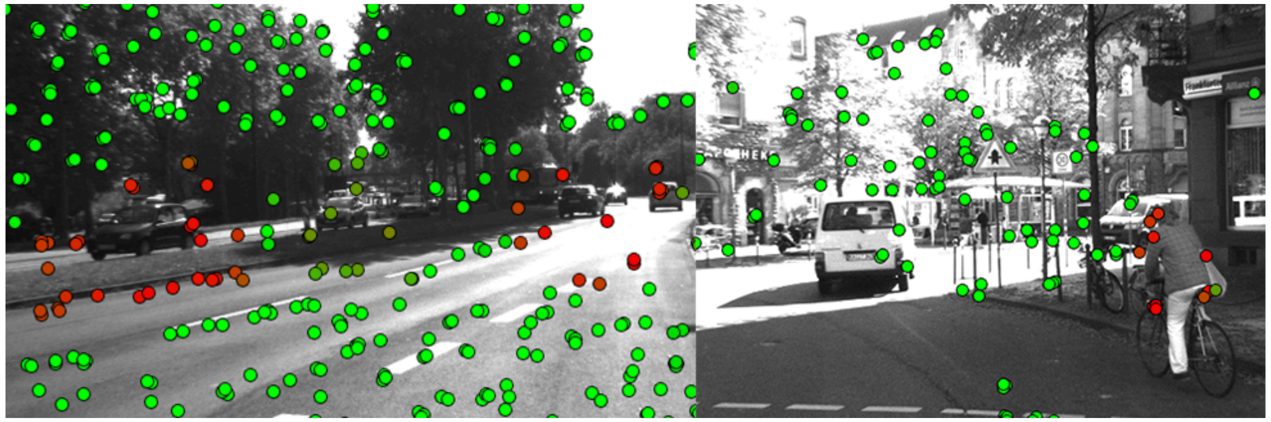
Predictive Robust Estimation



Latent Predictive Factors



Entropy



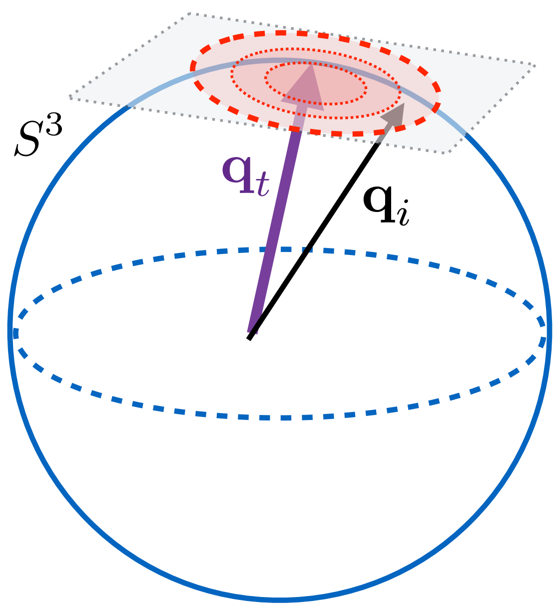
Optical Flow Variance

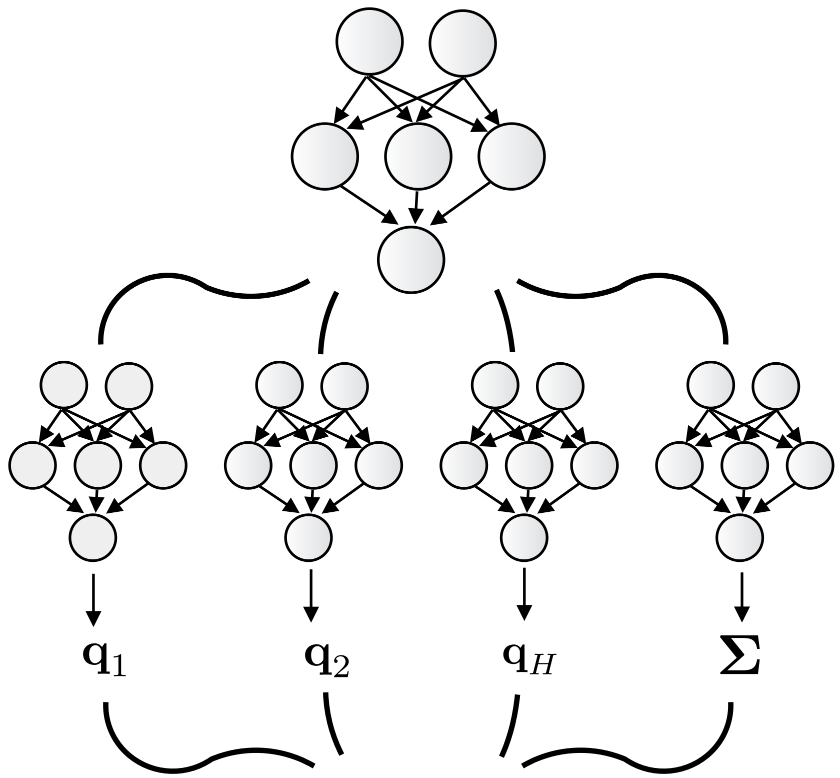
Covariance-aware **Loss**

- In order to learn a covariance matrix we define a supervised loss based on the tangent space of each target rotation:

$$\mathcal{L}_{\text{NLL}}(\mathbf{q}, \mathbf{q}_t, \mathbf{\Sigma}) = \boldsymbol{\phi}^T \mathbf{\Sigma}^{-1} \boldsymbol{\phi} + \log\det\left(\mathbf{\Sigma}\right)$$

with $\boldsymbol{\phi} = \operatorname{Log}\left(\mathbf{q} \otimes \mathbf{q}_t{}^{-1}\right)$





$$\mathcal{L} = \sum_{h=1}^{H} \mathcal{L}_{\text{NLL}}(\mathbf{q}_h, \mathbf{q}_t, \mathbf{\Sigma})$$

Hu and Kantor, "**Parametric Covariance Prediction for Heteroscedastic Noise**", **IROS** (2015)
Forster et al., "**IMU Preintegration on Manifold…**", **RSS** (2015)

Residual Learning in Robotics

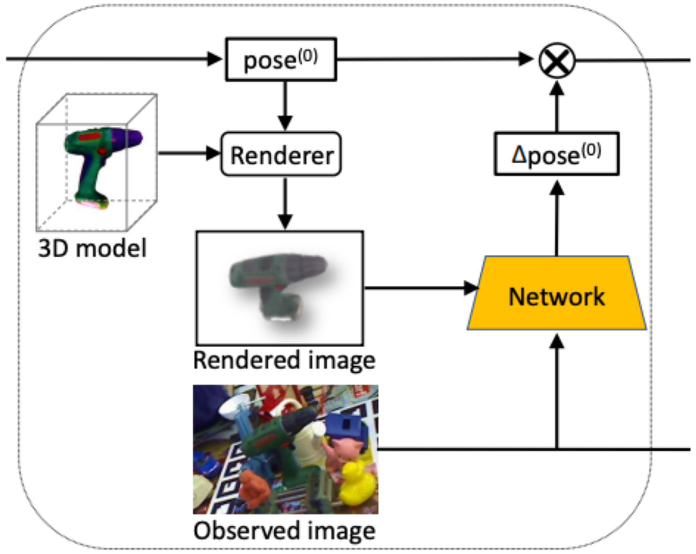
Helicopter Dynamics

Punjani and Abbeel, **Deep Learning Helicopter Dynamics Models**. **ICRA** (2015).

Throwing Objects

Zeng et al., **TossingBot: Learning to Throw Arbitrary Objects with Residual Physics**. **RSS** (2019).

Object Pose Regression



Li et al., **DeepIM: Deep Iterative Matching for 6D Pose Estimation**, **ECCV** (2018).

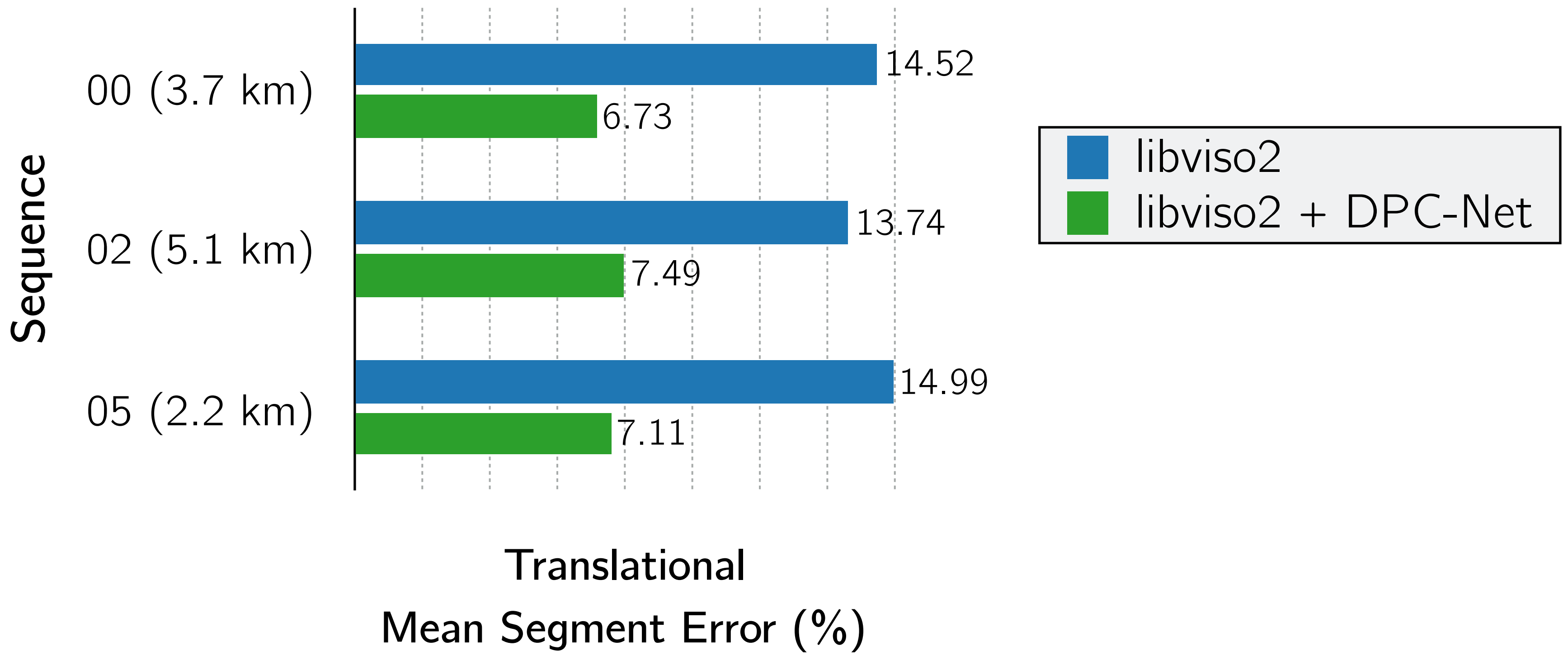
DPC-Net | Correcting for Lens Distortion

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = \left(1 + \kappa_1 r^2 + \kappa_2 r^4 + \kappa_3 r^6\right) \begin{bmatrix} x_n \\ y_n \end{bmatrix}$$

Original

Radially distorted

Distorted & cropped

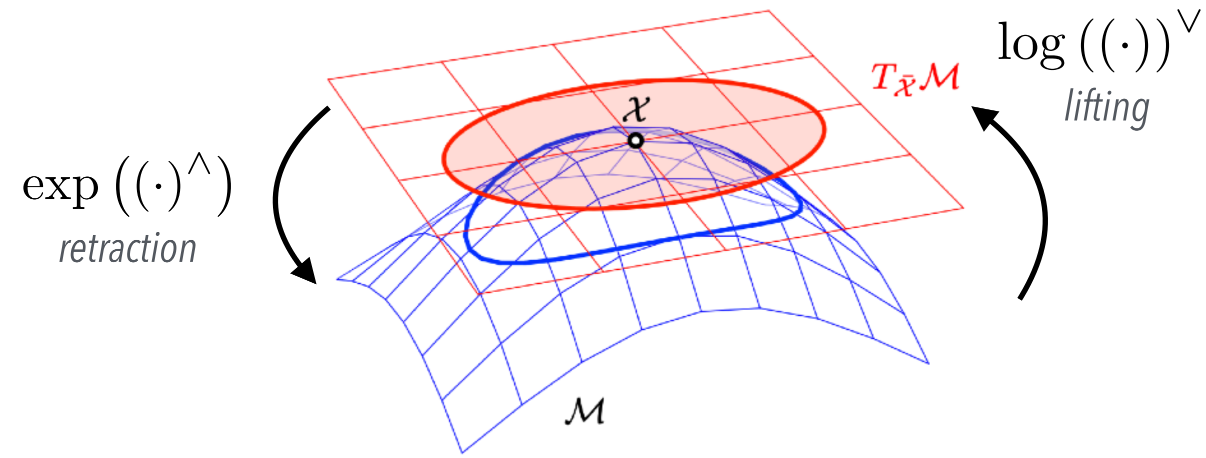


SO(3) and SE(3)

Rigid-body rotations form the matrix Lie group SO(3) - the Special Orthogonal group:

$$\mathrm{SO}(3) = \{\mathbf{R} \in \mathbb{R}^{3\times 3} |\ \mathbf{R}^T\mathbf{R} = \mathbf{1}, \det \mathbf{R} = 1\}$$

$$\mathbf{R} = \exp\left(\boldsymbol{\phi}^\wedge\right) = \sum_{n=0}^{\infty} \frac{1}{n!}(\boldsymbol{\phi}^\wedge)^n \qquad \boldsymbol{\phi} \in \mathbb{R}^3$$



Rigid-body *transformations* form the matrix Lie group SE(3) - the Special Euclidian group:

$$\mathrm{SE}(3) = \left\{\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \in \mathbb{R}^{4\times 4} |\ \mathbf{R} \in \mathrm{SO}(3), \mathbf{t} \in \mathbb{R}^3\right\}$$

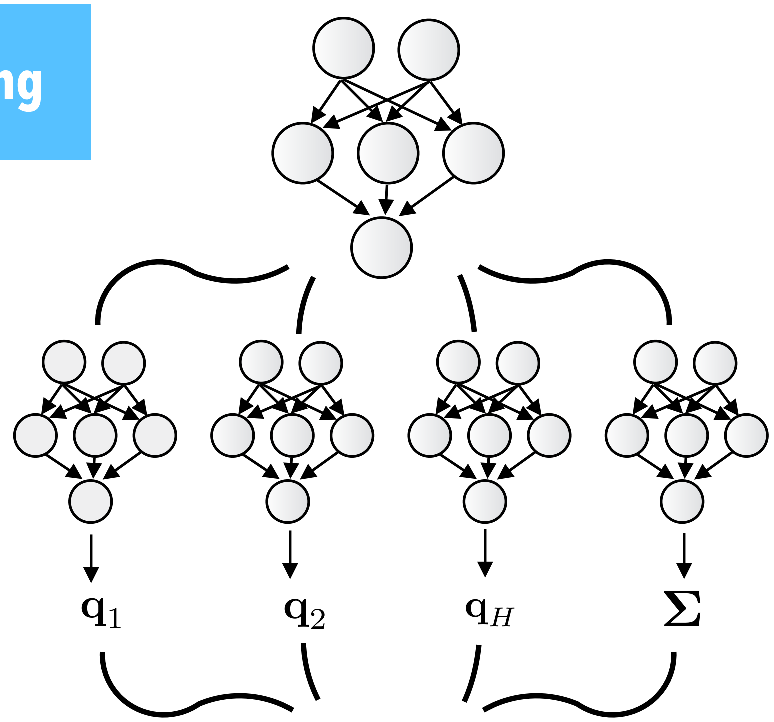
$$\mathbf{T} = \exp\left(\boldsymbol{\xi}^\wedge\right) = \sum_{n=0}^{\infty} \frac{1}{n!}(\boldsymbol{\xi}^\wedge)^n \qquad \boldsymbol{\xi} \in \mathbb{R}^6$$

$\boldsymbol{\phi}, \boldsymbol{\xi}$ are unconstrained, *but*
$\exp\left((\cdot)^\wedge\right)$ is **surjective**

Sola et al., "**A micro Lie theory for State Estimation in Robotics**", arXiv (2019)
Barfoot., "**State Estimation for Robotics.**" Cambridge University Press (2017)

Deep Probabilistic Regression of Rotations

Training

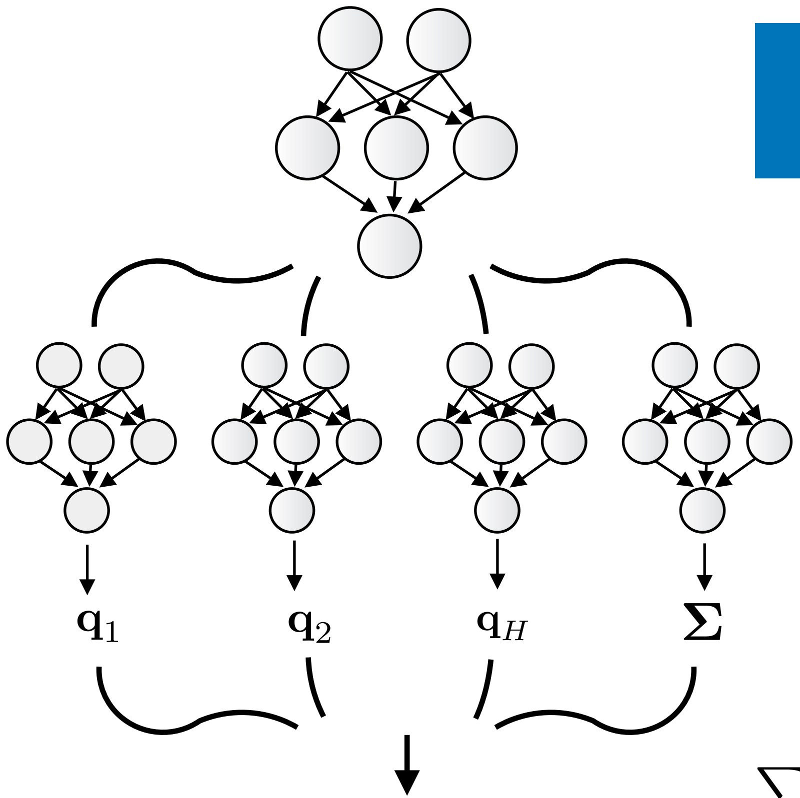


$$\mathcal{L} = \sum_{h=1}^{H} \mathcal{L}_{\text{NLL}}(\mathbf{q}_h, \mathbf{q}_t, \mathbf{\Sigma})$$

$$\mathcal{L}_{\text{NLL}}(\mathbf{q}, \mathbf{q}_t, \mathbf{\Sigma}) = \boldsymbol{\phi}^T \mathbf{\Sigma}^{-1} \boldsymbol{\phi} + \log\det(\mathbf{\Sigma})$$

$$\boldsymbol{\phi} = \operatorname{Log}\left(\mathbf{q} \otimes \mathbf{q}_t^{-1}\right)$$

Testing



1 Compute mean rotation

$$\overline{\mathbf{q}} = \frac{\sum_{h=1}^{H} \mathbf{q}_h}{\left\| \sum_{h=1}^{H} \mathbf{q}_h \right\|}$$

2 Compute residuals

$$\boldsymbol{\phi}_h = \operatorname{Log}\left(\mathbf{q}_h \otimes \overline{\mathbf{q}}^{-1}\right)$$

3 Compute covariance as

epistemic

$$\boxed{\frac{1}{H-1}\sum_{h=1}^{H} \boldsymbol{\phi}_h \boldsymbol{\phi}_h^T} + \boxed{\mathbf{\Sigma}}$$

aleatoric

DPC-Net | Loss

We use a Mahalanobis-like norm loss,

$$\mathcal{L}(\boldsymbol{\xi}) = \frac{1}{2} g(\boldsymbol{\xi})^T \boldsymbol{\Sigma}^{-1} g(\boldsymbol{\xi}) \quad \text{where} \quad g(\boldsymbol{\xi}) \triangleq \log\left(\exp\left(\boldsymbol{\xi}^\wedge\right) \mathbf{T}^{*\,-1}\right)^\vee$$

$\boldsymbol{\xi} \in \mathbb{R}^6$
network output

$\mathbf{T}^* \in \mathrm{SE}(3)$
target

Our loss has a covariance-based metric tensor that **naturally balances** rotation and translation terms:

$$\boldsymbol{\Sigma} = \frac{1}{N-1} \sum_{i=1}^{N} \left(\boldsymbol{\xi}_i^* - \overline{\boldsymbol{\xi}^*}\right)\left(\boldsymbol{\xi}_i^* - \overline{\boldsymbol{\xi}^*}\right)^T \quad \text{where} \quad \boldsymbol{\xi}_i^* \triangleq \log\left(\mathbf{T}_i^*\right)^\vee$$

training targets

compare to..

$$\mathcal{L} = \|\hat{\mathbf{x}} - \mathbf{x}\| + \beta \left\| \hat{\mathbf{q}} - \frac{\mathbf{q}}{\|\mathbf{q}\|} \right\|$$

Kendall et al., PoseNet, **ICRA** (2016)

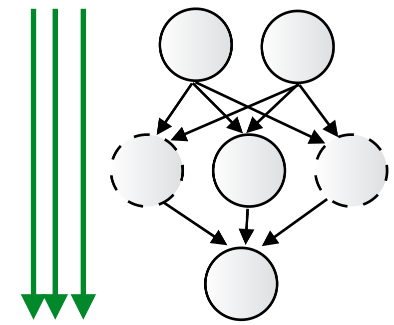
We derive an **analytic gradient** based on middle perturbations:

$$\frac{\partial \mathcal{L}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} = g(\boldsymbol{\xi})^T \boldsymbol{\Sigma}^{-1} \frac{\partial g(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} \qquad \frac{\partial g(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} = \boldsymbol{\mathcal{J}}\left(g(\boldsymbol{\xi})\right)^{-1} \boldsymbol{\mathcal{J}}(\boldsymbol{\xi}) \quad \text{where}$$

$\boldsymbol{\mathcal{J}}(\cdot)$
left SE(3) Jacobian

Monte Carlo Dropout Revisited

Monte Carlo Dropout
(Variational Inference)



$\overline{y}, \mathrm{var}(\{y_i\}) + \tau^{-1}$

Gal, "**Uncertainty in Deep Learning**" **Ph.D. Thesis** (2016)

- To make connection between dropout and Bayesian NNs, Gal turns to **variational inference** to approximate posterior over weights:

$$q(\boldsymbol{w}) \sim p(\boldsymbol{w}|\mathbf{X}, \mathbf{S})$$

(training images) (training targets)

(matrix with K_i weights *for* layer i)

$$q(\boldsymbol{w}_i) = \mathbf{M}_i \ \mathrm{diag}\left\{ \{b^i_j\}_{j=1}^{K_i} \right\},$$

$$b^i_j \in \mathrm{Bernoulli}(p_i)$$

(dropout probability)

- Gal shows that the **dropout training loss is equivalent to minimizing the KL divergence** between the posterior and this variational distribution

$$D_{\mathrm{KL}} \left(\ p(\boldsymbol{w}|\mathbf{X}, \mathbf{S}) \ || \ q(\boldsymbol{w}) \ \right)$$

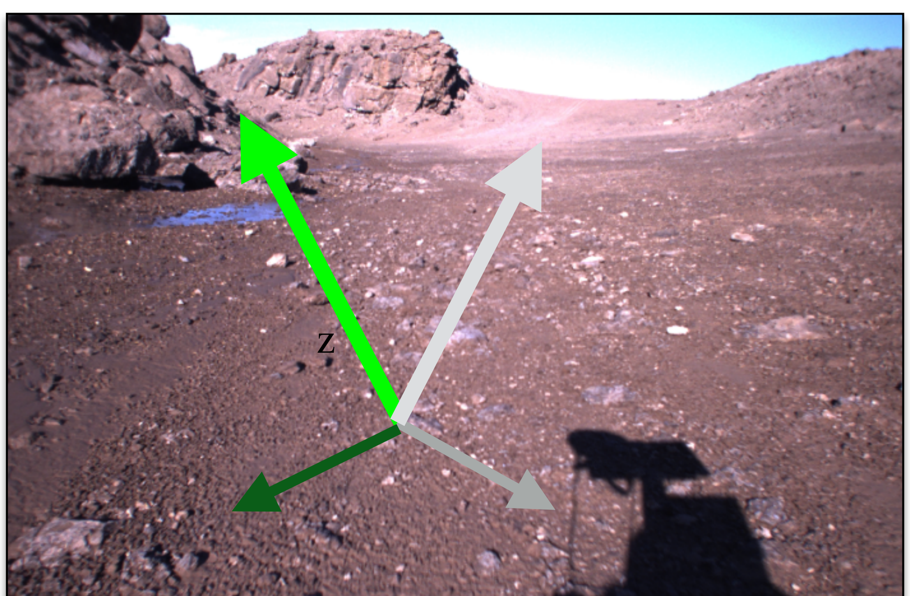
- **However**, for linear networks, one can show that this posterior **does not concentrate with more data**, and requires careful tuning of hyper-parameters

Osband, "**Risk versus uncertainty in deep learning: Bayes, bootstrap and the dangers of dropout,**" **NeurIPS** (2017)

Sun-BCNN Training

KITTI & Devon Island

KITTI Vision Benchmark

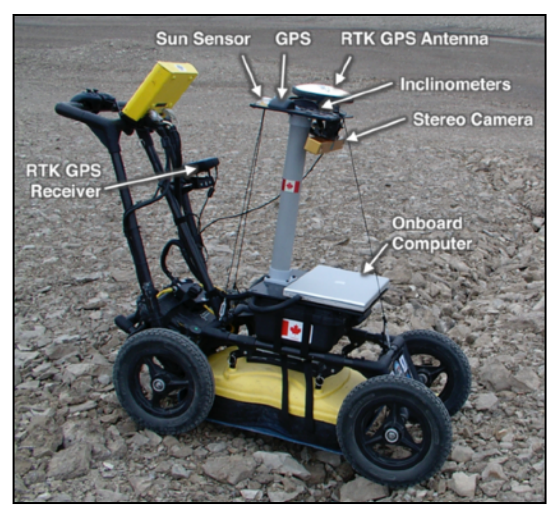


Devon Island Rover Navigation

Ground truth
6-DoF GPS-INS and a solar ephemeris model (based on GPS timestamp)

Cosine distance loss

$$\mathcal{L}(\hat{\mathbf{s}}_k, \mathbf{s}_k) = 1 - \hat{\mathbf{s}}_k \cdot \mathbf{s}_k$$



Ground truth
Sinclair Interplanetary digital sun sensor

Rotation Averaging

$$\overline{\mathbf{R}} = \underset{\mathbf{R} \in SO(3)}{\operatorname{argmin}} \sum_{i=1}^{n} d(\mathbf{R}_i, \mathbf{R})^2$$

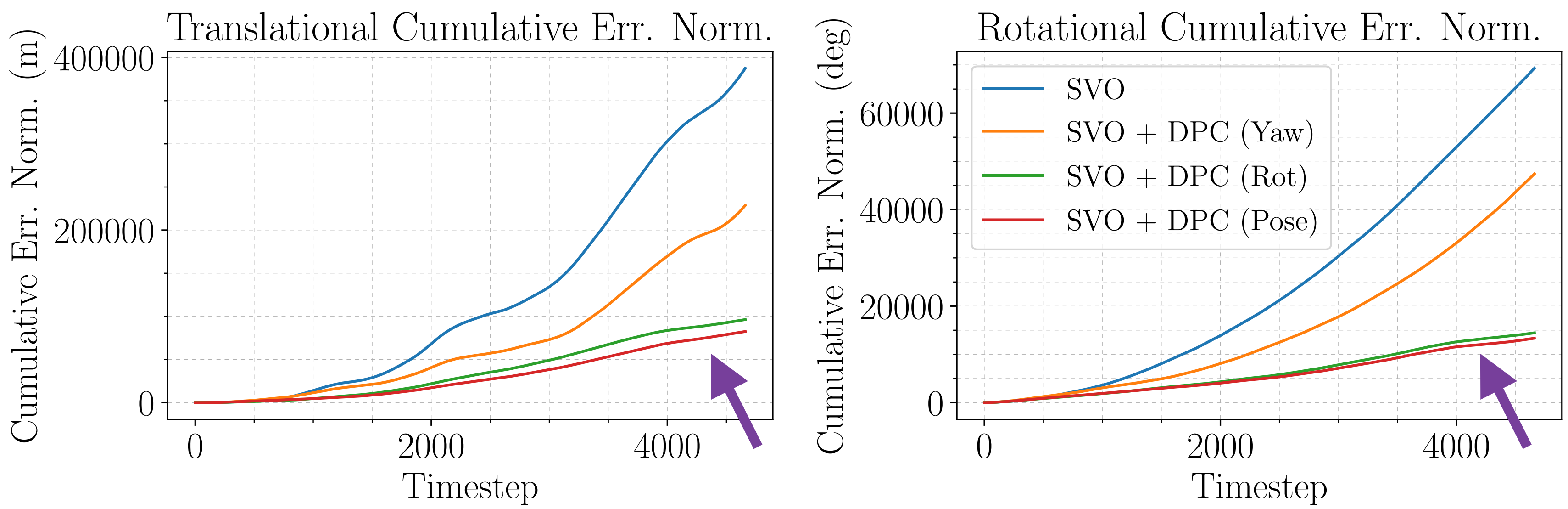
	Metric	Resulting Mean
Angular *(Geodesic)*	$d_{\text{ang}}(\mathbf{R}_a, \mathbf{R}_b) = \left\| \operatorname{Log}\left(\mathbf{R}_a \mathbf{R}_b^T\right) \right\|_2 = \theta$	Karcher mean **(requires iteration)**
Chordal	$d_{\text{chord}}(\mathbf{R}_a, \mathbf{R}_b) = \left\| \mathbf{R}_a - \mathbf{R}_b \right\|_{\text{Frob}} = 2\sqrt{2} \sin \frac{\theta}{2}$	Euclidian mean in R9 projected onto SO(3) **(requires SVD)**
Quaternion	$d_{\text{quat}}(\mathbf{q}_a, \mathbf{q}_b) = \min\left(\left\| \mathbf{q}_a - \mathbf{q}_b \right\|_2, \left\| \mathbf{q}_a + \mathbf{q}_b \right\|_2 \right) = 2 \sin \frac{\theta}{4}$	Arithmetic mean projected onto unit sphere **(simple analytic expression)**

Hartley et al., "**Rotation Averaging**", **IJCV (2013)**

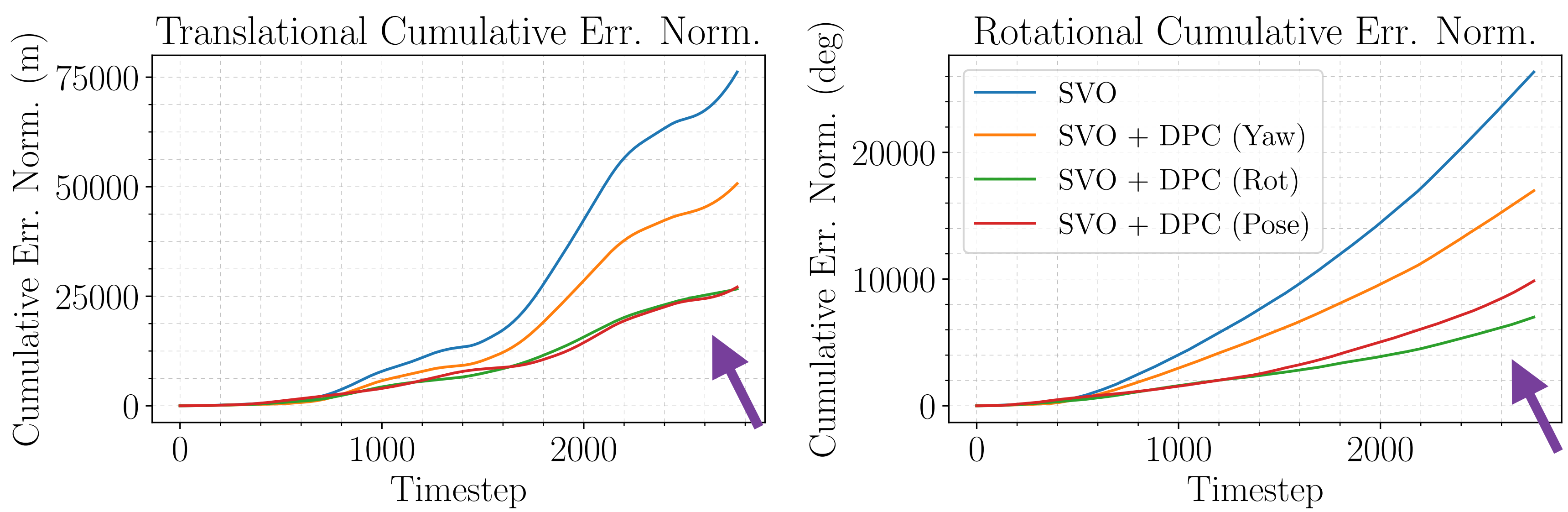
DPC-Net | Rotation Corrections Only?

- Correcting **rotation only** can be nearly as good as correcting full pose corrections
- Metric translation information is **difficult to generalize**

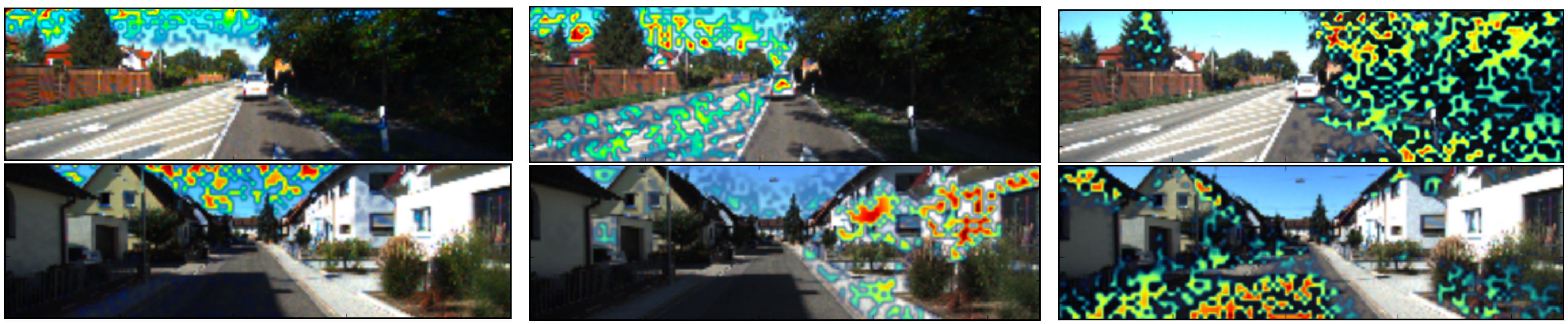
Sequence **02** (2.2 km)



Sequence **05** (2.2 km)



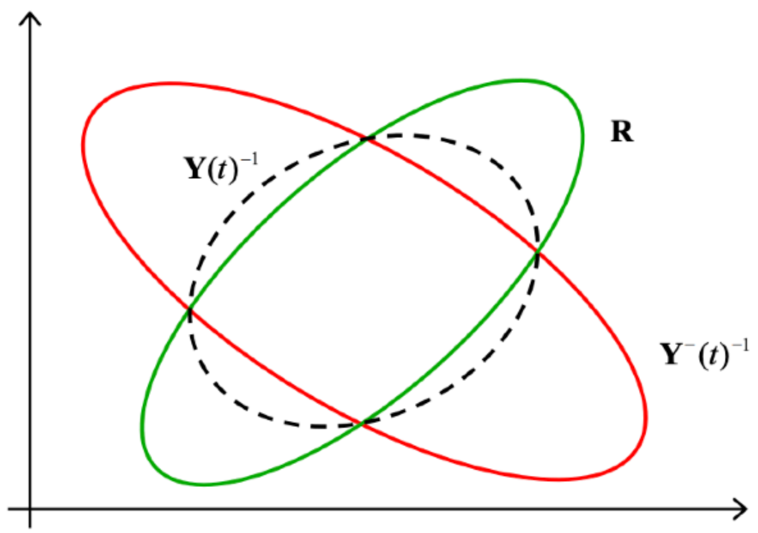
Sun-BCNN ‘activations’

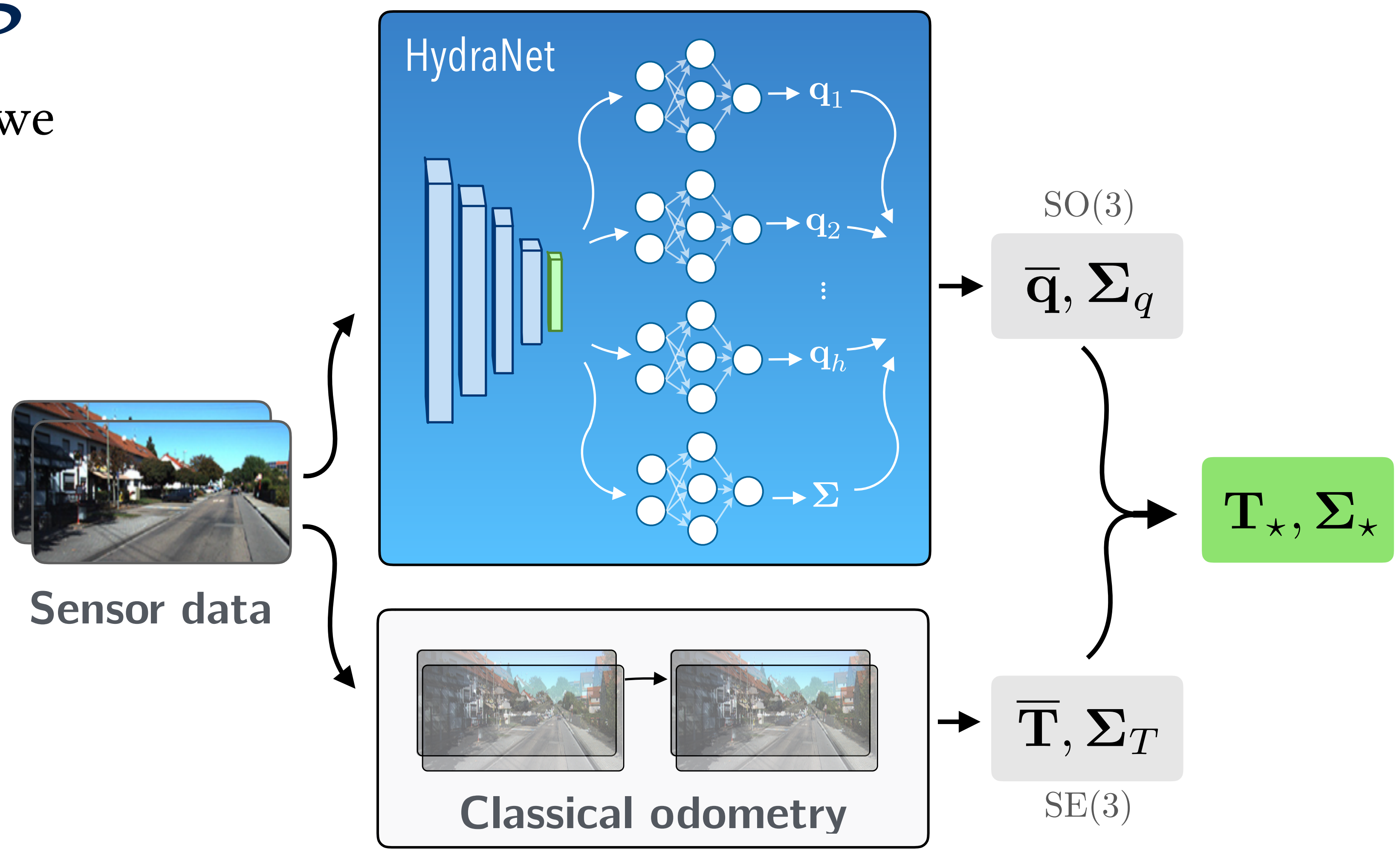


Sky Well lit regions Shadows

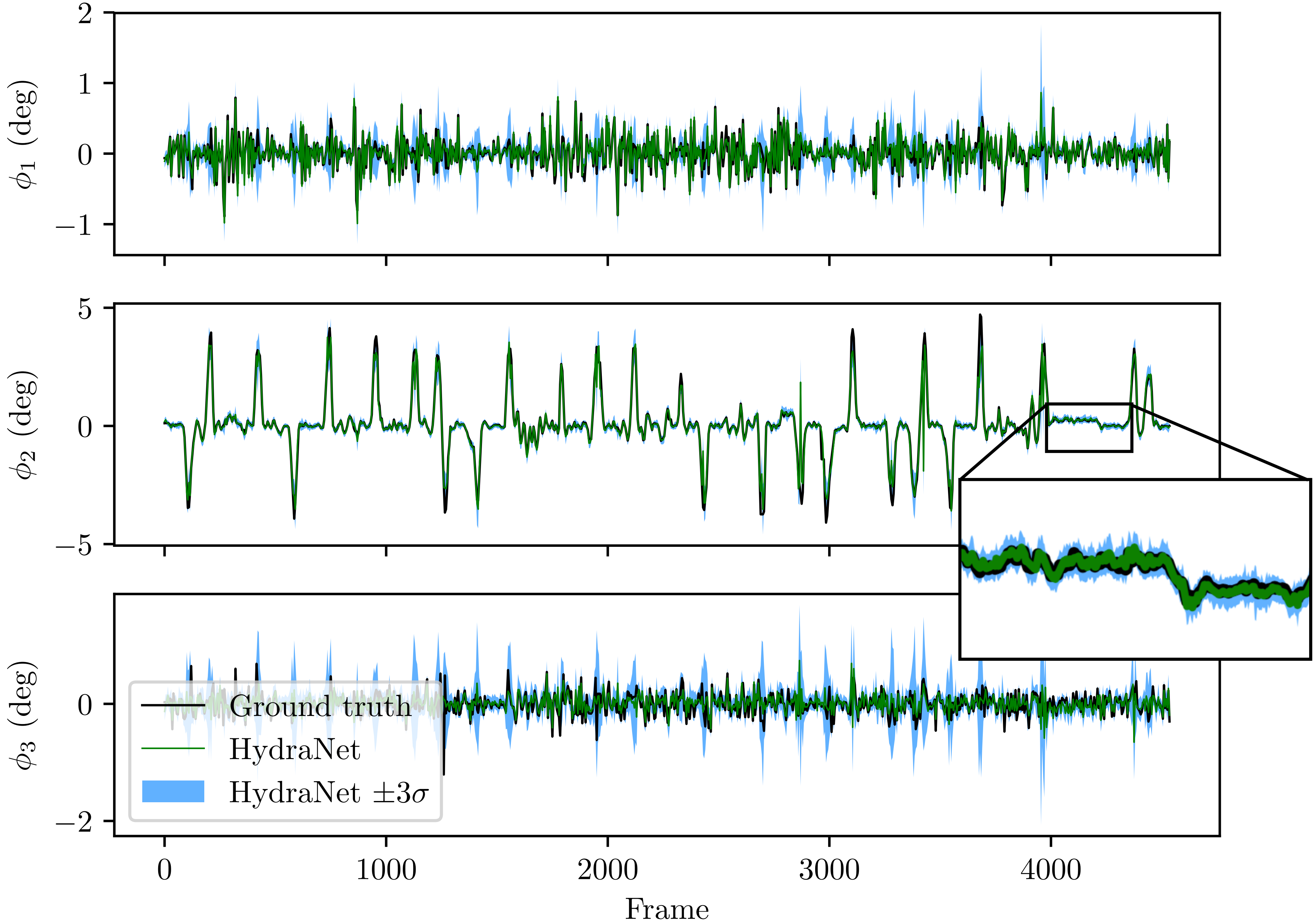
Double Counting

- In order to mitigate correlations, we are investigating to *Covariance Intersection*

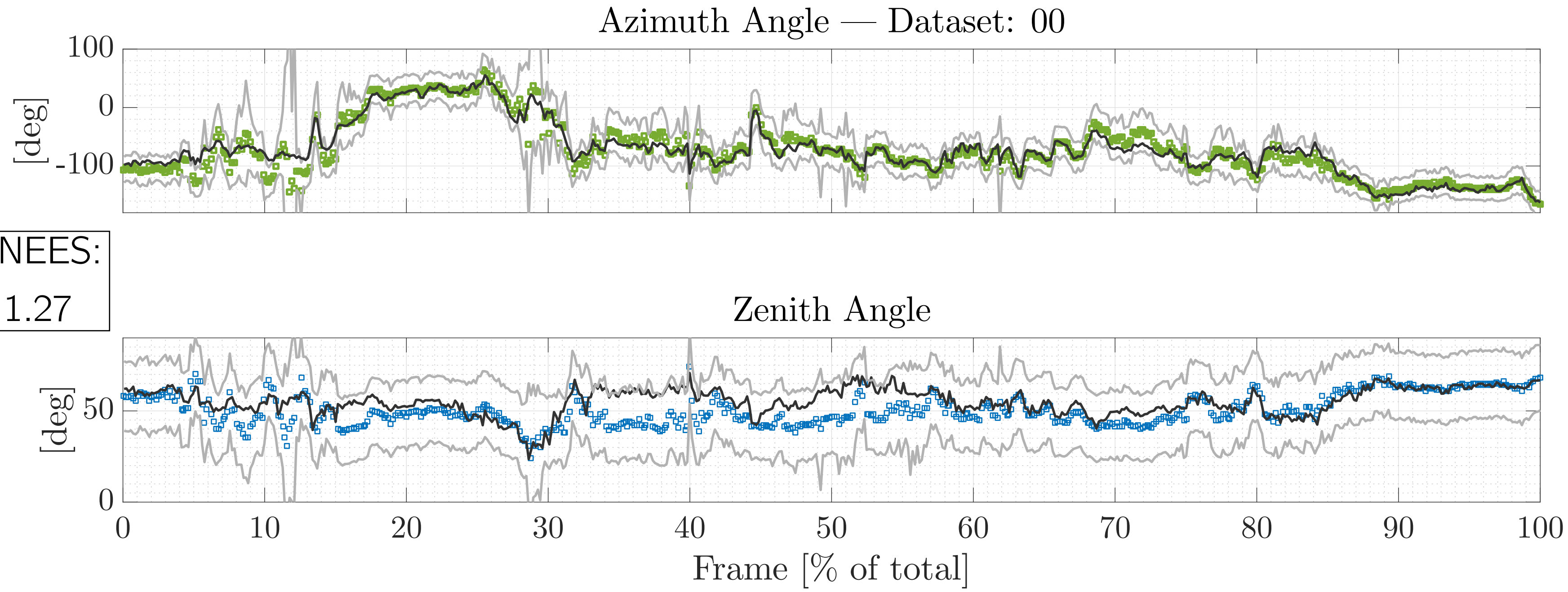




HydraNet Predictions

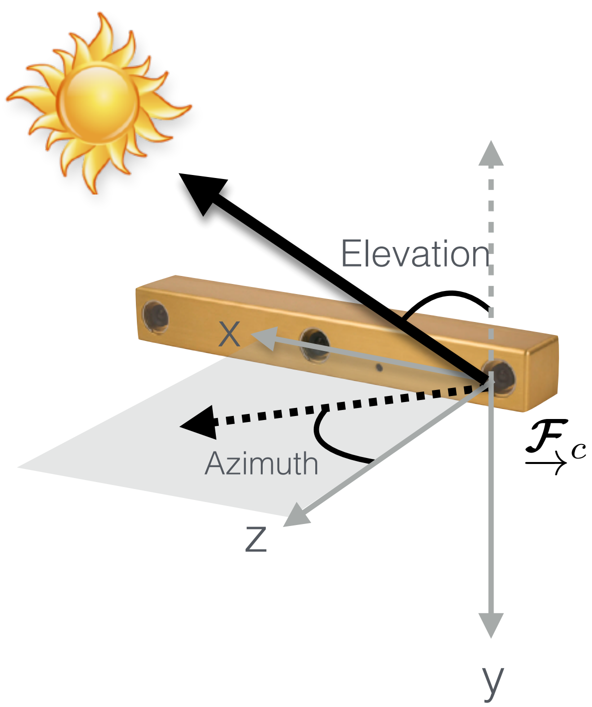


Visual sun sensing: Devon Island

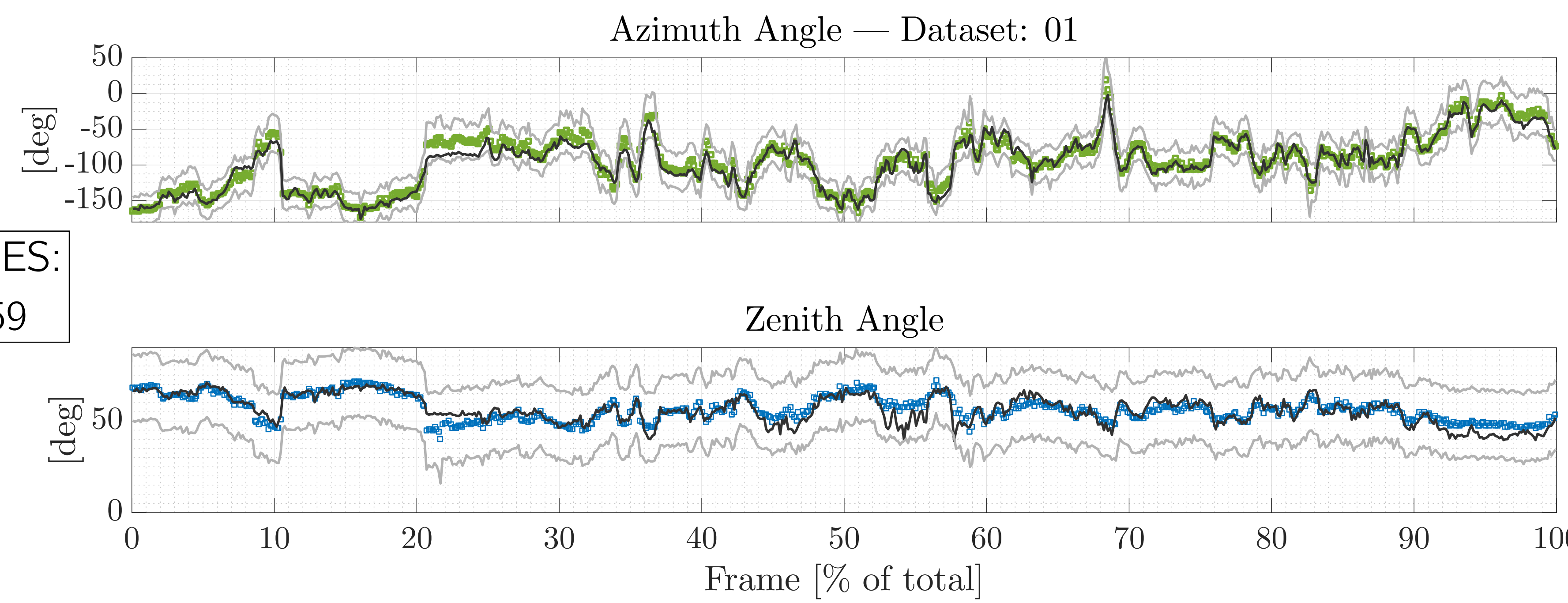


ANEES: 1.27

Predicted **Azimuth** Angle
Predicted **Zenith** Angle
Ground Truth (Sun Sensor)
Predicted **+/- 3 sigma**

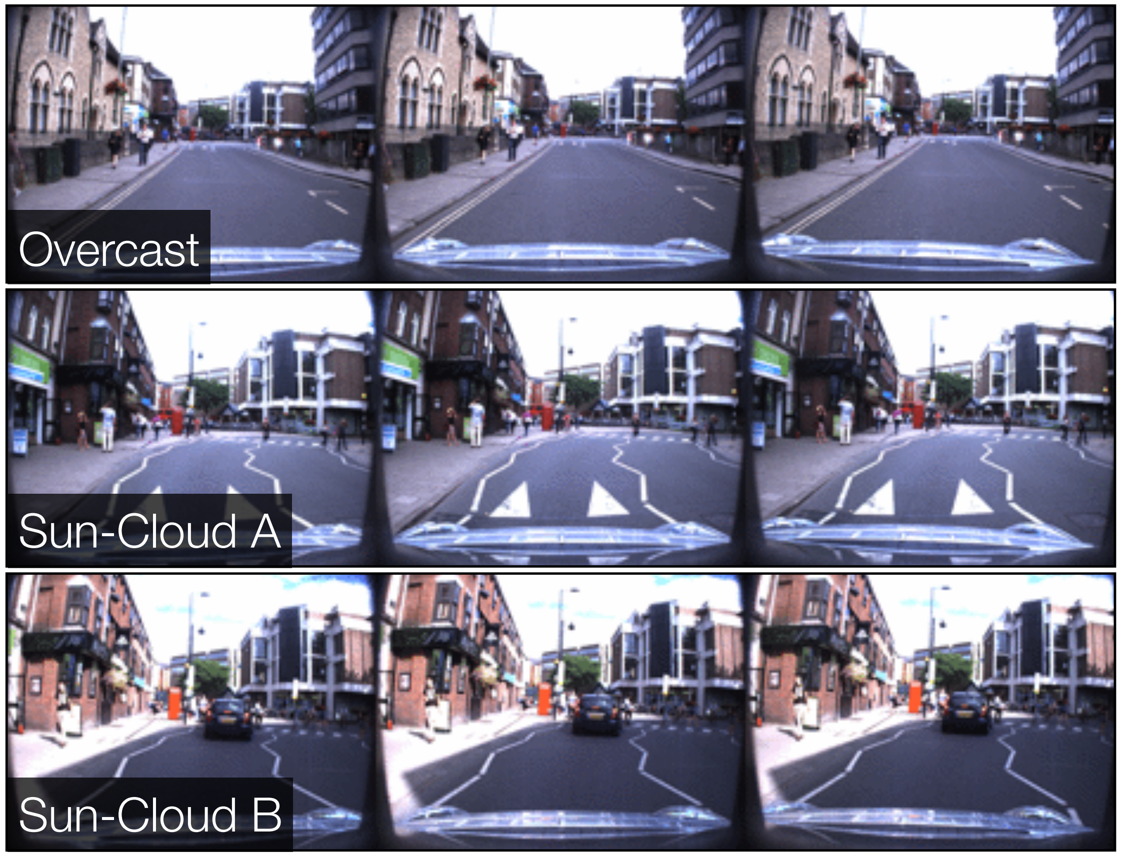


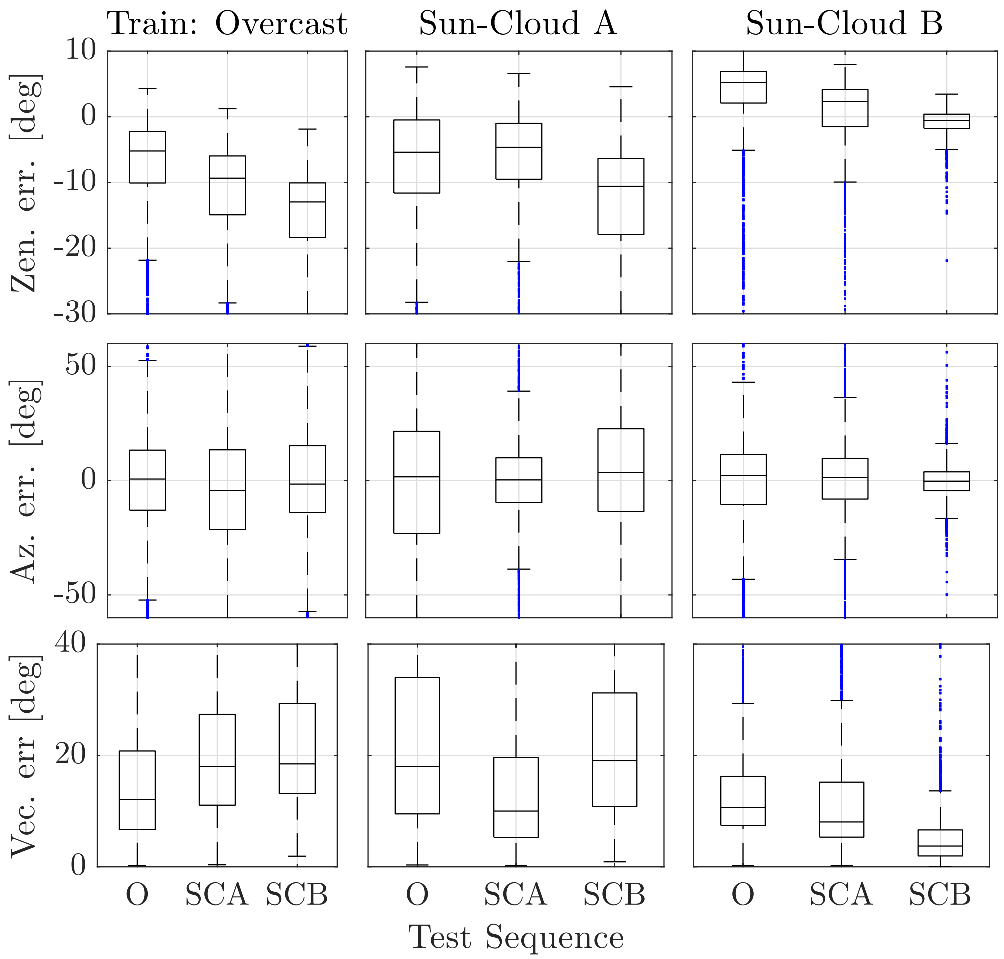
ANEES: 0.59



Sensitivity analysis

We tested the effect of cloud cover using the ***Oxford Robotcar Dataset***





Sun-BCNN **works in cloudy conditions**, especially when trained in sunny conditions

Maddern et al., *1 year, 1000 km: The Oxford RobotCar dataset*, **IJRR (2016)**

DPC-Net | Improving stereo VO

- We reduce the m-ATE (mean Absolute Trajectory Error) of our estimator by up to **75%**