DynaNet: Neural Kalman Dynamical Model for Motion Estimation and Prediction

Changhao Chen, Chris Xiaoxuan Lu, Bing Wang, Niki Trigoni, Andrew Markham
{firstname.lastname}@cs.ox.ac.uk
Department of Computer Science, University of Oxford

Abstract

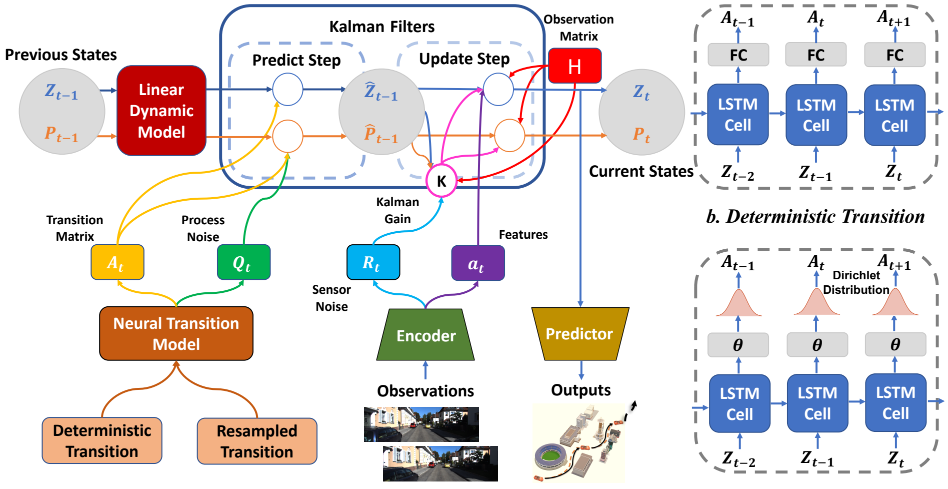
Dynamical models estimate and predict the temporal evolution of physical systems. State Space Models (SSMs) in particular represent the system dynamics with many desirable properties, such as being able to model uncertainty in both the model and measurements, and optimal (in the Bayesian sense) recursive formulations e.g. the Kalman Filter. However, they require significant domain knowledge to derive the parametric form and considerable hand-tuning to correctly set all the parameters. Data driven techniques e.g. Recurrent Neural Networks have emerged as compelling alternatives to SSMs with wide success across a number of challenging tasks, in part due to their ability to extract relevant features from rich inputs. They however lack interpretability and robustness to unseen conditions. In this work, we present DynaNet, a hybrid deep learning and time-varying state-space model which can be trained end-to-end. Our neural Kalman dynamical model allows us to exploit the relative merits of each approach. We demonstrate state-of-the-art estimation and prediction on a number of physically challenging tasks, including visual odometry, sensor fusion for visual-inertial navigation and pendulum control. In addition we show how DynaNet can indicate failures through investigation of properties such as the rate of innovation (Kalman Gain).

1 Introduction

From catching a ball to tracking the motion of planets across the celestial sphere the ability to estimate and predict the future trajectory of moving objects is key for interaction with our physical world. With ever increasing automation e.g. self-driving vehicles and mobile robotics, the ability to not only estimate system states based on sensor data, but also to reason about latent dynamics and therefore predict states with partial or even without any observation is of huge importance to safety and reliability [41].

Newtonian/Classical Mechanics has been developed as an explicit mathematical model which can be used to predict future motion and infer how an object has moved in the past. This is commonly captured in a State Space model (SSM) that describes the temporal relationship and evolution of states through first-order difference equations. For example, in the visual ego-motion estimation task, also known as Visual Odometry (VO) [35, 10, 12], velocity, position, and orientation are usually used as physically attributable states for mobile agents. Models are typically hand-crafted based on domain knowledge and require significant expertise to develop and tune. Simplifying assumptions are often made e.g. to treat the system as being linear, time-invariant with uncertainty being additive and Gaussian. A canonical example of an optimal Bayesian filter for linear systems is the Kalman Filter [21], which is an optimal linear quadratic estimator. Although capable of controlling sophisticated mechanical systems (e.g. the Apollo 11 lander used a 21 state Kalman Filter [32]), it becomes more challenging to use in complex, nonlinear systems, giving rise to alternative variants such as the Sequential Monte Carlo [28] or nonlinear graph optimisation [25]. However, even when using these

Machine Learning for Autonomous Driving Workshop at the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.



a. Our Proposed DynaNet Framework

b. Deterministic Transition

c. Resampled Transition

Figure 1: The DynaNet framework (a) consists of the neural observation model to extract latent states $\mathbf{z}$, the neural transition model to generate the evolving relation $\mathbf{A}$, and a recursive Kalman Filter to infer and predict system states. Two transition generation strategies - (b) deterministic transition or (c) resampled transition are proposed to achieve desired system behaviours.

sophisticated approaches, imperfections in model parameters and measurement errors from sensory data contribute to issues such as accumulative drift in visual navigation systems. Furthermore, there is a disconnect between the complexities of rich sensor data e.g. images and derived states.

Identifying the underlying mechanism governing the motion of an object is a hard problem to solve for dynamical systems operating in real world. As a consequence, in recent years, there has been an explosive growth in applying deep neural networks (DNNs) for motion estimation [5, 44, 47, 6, 29, 1, 3, 46, 45]. These learning based approaches can extract useful representations from high-dimensional raw data to estimate the key motion states in an end-to-end fashion. Although these learned models demonstrate good performance in both accuracy and robustness, they are 'black-boxes' regressing measurements with implicit latent states and difficult to interpret. In contrast to neural networks, state-space-models (SSMs) are able to construct a parametric model description and offer an explicit transition relation that describes the evolution of system states and uncertainty into the future. They can also optimally fuse measurements from multiple sensors based on their innovation gain, rather than simply stacking them as in a neural network.

In this work, we propose **DynaNet** - neural Kalman dynamical models, combining the respective advantages of deep neural networks (powerful feature representation) and state space models (explicit modelling of physical processes). As illustrated in Figure 1, our proposed end-to-end model enables automatic feature extraction from raw data and constructs a time-varying transition model. The recursive Kalman Filter is incorporated to provide optimal filtering on the feature state-space and estimate process covariance to reason about system behaviour and stability. This allows for accurate system identification in an end-to-end manner without human effort.

Specifically, our model benefits from: (1) A novel approach that can learn a linear-like state-space model directly from raw data. (2) A Kalman filter is adopted on the feature level for latent state inference. It works efficiently across a number of challenging situations e.g. when only partial/corrupted observations are available or even without any observations. 3) By resampling the transition matrix from a Dirichlet distribution, the neural system could be ensured asymptotically stable. 4) The explicit parameters inside the state-space-model show an interpretability of system behaviours, for example, kalman gain indicates observation failures.

To demonstrate the effectiveness of the proposed technique, we have conducted extensive experiments and a systematic study on challenging real world motion estimation and prediction tasks, including pendulum control, visual odometry and visual-inertial odometry. We show how the proposed method outperforms state-of-the-art deep-learning techniques, whilst yielding valuable information about model performance.

2 Neural Kalman Dynamic Models

We consider a time-dependent dynamical system, governed by a complex evolving function f:

$$\mathbf{z}_t = f(\mathbf{z}_{t-1}, \mathbf{w}_t) \tag{1}$$

where $\mathbf{z} \in \mathbb{R}^d$ is d-dimensional latent state, t is the current timestep, and $\mathbf{w}$ is a random variable capturing system and measurement noise. The evolving function f is assumed to be Markovian, describing the state-dependent relation between latent states $\mathbf{z}_t$ and $\mathbf{z}_{t-1}$. The model in Equation (1) can be reformulated as a linear-like structure, i.e. the state-dependent coefficient (SDC) form, with a time-varying transition matrix $\mathbf{A}$:

$$\mathbf{z}_t = \mathbf{A}_t \mathbf{z}_{t-1} \tag{2}$$

Notably, system nonlinearity is not restricted by this linear-like structure [4], as there always exists a SDC form $f(z) = A(z)z$ to express any continuous differentiable function f with $f(0) = 0$. In this regard, our problem of the dynamic model is how to recover the latent states $\mathbf{z}$, and their time-varying transition relation $\mathbf{A}$ from high-dimensional measurements $\mathbf{x}$ (e.g. a sequence of images) without resort to a hand-crafted physical model. This work aims to construct and reparameterize this dynamic model using the expressive power of deep neural networks. Figure 1 shows the main framework, which is discussed in detail in the following sections.

To avoid confusion, in the rest of this paper, *latent* states $\mathbf{z}$ are exclusively used for dynamical models and *hidden* states $\mathbf{h}$ exclusively represent the neurons in a deep neural network.

2.1 Neural Emission Model

Intuitively, the system states containing useful information often lie in a different latent space different to the original measurements. For example, given a sequence of images (the measurements), the key system states of visual odometry are velocity, orientation and position. However, it is non-trivial for conventional models to formulate a temporal linear model that can precisely describe the relation between these physical representations [27]. Rather than explicitly specifying physical states as in a classical SSM, we use a deep network to automatically extract latent state features and constrain these through optimisation to follow the linear-like relation in Equation (2). Importantly, this linearization allows us to use a Kalman Filter directly for state feature inference.

In our neural emission model, an encoder f_{encoder} is used to extract both features $\mathbf{a}_t$ and an estimation of uncertainty $\boldsymbol{\sigma}_t^a$ from the data $\mathbf{x}_t$ at timestep t:

$$\mathbf{a}_t, \boldsymbol{\sigma}_t = f_{\text{encoder}}(\mathbf{x}_t). \tag{3}$$

The features $\mathbf{a}$ act as observations of the latent feature state space. The coupled uncertainties $\boldsymbol{\sigma}$ represent the measurement belief that can be used as the observation noise $\mathbf{R}$ in a Kalman Filter. However, the observations $\mathbf{a}$ are sometimes unable to provide sufficient information for all latent states $\mathbf{z}$ in a dynamical system, for example, the occasional absences of sensory data. Hence, a deterministic emission matrix $\mathbf{H}$ is defined to connect with the full latent states $\mathbf{z}$:

$$\mathbf{a}_t = \mathbf{H}\mathbf{z}_t \tag{4}$$

In a practical setting, when the extracted features contain all the information for dynamical systems, the emission matrix $\mathbf{H}$ is set to d-dimensional identity matrix $\mathbf{I}_d$ as features and states are identical at this moment. On the other side, the identity matrix needs to adapt to $\mathbf{H} = [\mathbf{I}_m, \mathbf{0}_{m\times(d-m)}]$, when observations only give rise to m features. In this case, the rest $(d - m)$ latent states will be attained from historical states. Our experiment in Section 3.3 demonstrates the superiority this neural emission model in addressing the issue of observation absences for sensor fusion in visual-inertial odometry.

2.2 Neural Transition Model

In a SSM, the temporal evolution of latent states is determined by the transition matrix $\mathbf{A}$ as in Equation 2. Obviously, the transition matrix is of considerable importance as it directly describes the governing mechanism of a system. Nevertheless, such a matrix is difficult to specify manually, especially when it is time-varying. Figure 1 demonstrates two methods we propose to estimate $\mathbf{A}$ on the fly, based on prior system states: (1) a deterministic way to learn it end-to-end from raw data; (2) a stochastic way to resample it from a distribution, e.g. the Dirichlet distribution in this work.

2.2.1 Deterministic Transition

Intuitively, movement change depends on historical system states, which are encoded in the latent states $\mathbf{z}_{0:t-1}$. Previous work mostly applied a recurrent neural network (RNN) to specify dynamic weights for choosing and interpolating between a fixed number of different transition modes [22, 13]. Inspired by [38], our model generates the transition matrix directly from the latent states $\mathbf{z}$. In our neural transition model, the dependence of the transition matrix on historical latent states is specified by a RNN. This RNN recursively processes previous hidden states $(\mathbf{z}_{t-1}, \mathbf{h}_{t-1})$ of the dynamic model and Long Short-Term Memory (LSTM) cell respectively, and outputs the time-dependent transition matrix $\mathbf{A}_t$:

$$\mathbf{A}_t = \text{LSTM}(\mathbf{z}_{t-1}, \mathbf{h}_{t-1}). \tag{5}$$

2.2.2 Resampled Transition

The linear-like SSM structure in Equation (2) allows for a quadratic Lyapunov stability analysis, whilst advances in stochastic optimisation allow to construct neural probabilistic models [39]. We thus propose to resample the transition matrix from a predefined probability distribution to enforce a desired property, namely stability. Based on the findings in [43], if the state transition follows a Dirichlet distribution in a positive system, it will lead to a model being asymptotically stable i.e. it will be Bounded Input Bounded Output (BIBO) stable. Constructing the stochastic variable and determining the parameters of the distribution are easy to achieve with the widely used reparameterisation trick [19].

Specifically, in this work, the concentration $\boldsymbol{\alpha}$ of the Dirichlet distribution is generated from historical system states via an LSTM based RNN:

$$\boldsymbol{\alpha} = \text{LSTM}(\mathbf{z}_{t-1}, \mathbf{h}_{t-1}) \tag{6}$$

A small Gaussian random noise is also added to improve model robustness. At each timestep, a realisation of the transition matrix $\mathbf{A}$ is drawn from the constructed Dirichlet distribution:

$$\mathbf{A}_t \sim \text{Dirichlet}(\boldsymbol{\alpha}) \tag{7}$$

Note that in DynaNet the transition states are in the latent feature rather than the final target states, i.e. VO states. The latent features are extracted by the encoder, which ensures the transition strictly positive through ReLU activation and a tiny random positive number on the last layer of the DNN based encoder.

2.3 Prediction and Inference with a Kalman Filter

The neural emission model estimates system states from noisy sensor measurements, while the generated transition model describes the system evolution and predicts the system states with previous ones. However, uncertainties exist in both of them and motivate us to integrate a Kalman Filter into our framework. The Kalman Filter recursively deals with the uncertainties, and produces a weighted average of the state predictions and fresh observations. With the aforementioned neural emission and transition models, the prediction and inference are performed on the feature state space and follow a standard Kalman filtering pipeline. We also note that the Kalman Filter's gain controls how much to update the residual error (i.e., the difference between prediction and observation), which is a useful metric to represent the relative quality of measurements (as shown in Section 3.5).

More specifically, the Kalman Filter consists of two blocks: prediction and update. In the prediction stage, prior estimates of the mean value and covariance $(\mathbf{z}_{t|t-1}, \mathbf{P}_{t|t-1})$ at the current timestep are derived from the posterior state estimates $(\mathbf{z}_{t-1|t-1}, \mathbf{P}_{t-1|t-1})$ in the previous timestep:

$$\begin{aligned} \mathbf{z}_{t|t-1} &= \mathbf{A}_t \mathbf{z}_{t-1|t-1}, \\ \mathbf{P}_{t|t-1} &= \mathbf{A}_t \mathbf{P}_{t-1|t-1} \mathbf{A}_t^T + \mathbf{Q}_t. \end{aligned} \tag{8}$$

When current observations $\mathbf{a}_t$ are available, the update process allows us to produce a posterior mean and covariance of hidden states $(\mathbf{z}_{t|t}, \mathbf{P}_{t|t})$ as follows:

$$\begin{aligned} \mathbf{r}_t &= \mathbf{a}_t - \mathbf{H}_t \mathbf{z}_{t|t-1}, \\ \mathbf{S}_t &= \mathbf{R}_t + \mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t^T, \\ \mathbf{K}_t &= \mathbf{P}_{t|t-1} \mathbf{H}_t^T \mathbf{S}_t^{-1}, \\ \mathbf{z}_{t|t} &= \mathbf{z}_{t|t-1} + \mathbf{K}_t \mathbf{r}_t, \\ \mathbf{P}_{t|t} &= (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{P}_{t|t-1}, \end{aligned} \tag{9}$$

where $\mathbf{r}$ is the residual error (aka. innovation), $\mathbf{S}$ is the residual covariance and $\mathbf{K}$ is Kalman gain. In contrast to hand-tuning process noise $\mathbf{Q}$ and measurement noise $\mathbf{R}$ in a conventional KF, these two terms are jointly learned by our proposed neural dynamical model. Finally, the predictor (e.g. a FC network) outputs the target values $\mathbf{y}_t$ from the estimated optimal hidden states $\mathbf{z}_{t|t}$:

$$\tilde{\mathbf{y}}_t = f_{\text{predictor}}(\mathbf{z}_{t|t}). \tag{10}$$

In the case that current measurements i.e. $\mathbf{a}_t$ are unavailable, the reconstructed values $\hat{\mathbf{y}}_t$ are inferred from the prior estimate $\mathbf{z}_{t|t-1}$:

$$\hat{\mathbf{y}}_t = f_{\text{predictor}}(\mathbf{z}_{t|t-1}). \tag{11}$$

All parameters θ in our model are end-to-end trained with a mean square loss function. This loss function jointly compares the ground truth $\mathbf{y}_t$ with posterior predictions $\tilde{\mathbf{y}}_t$ and prior prediction $\hat{\mathbf{y}}_t$:

$$L(\theta) = \frac{1}{T}\sum_{t=1}^{T}(||\mathbf{y}_t - \tilde{\mathbf{y}}_t||^2 + ||\mathbf{y}_t - \hat{\mathbf{y}}_t||^2) \tag{12}$$

3 Experiments

We systematically evaluate our system through extensive experiments including (1) Section 3.2 - pose estimation and prediction for visual odometry, (2) Section 3.3 - sensor fusion and prediction with partial observations for visual-inertial odometry and (3) Section 3.4 - learning with control signals for pendulum prediction. Moreover, an interpretability study is also conducted in Section 3.5.

3.1 Experiment Setup and Datasets

KITTI Odometry Dataset [15] is a commonly used benchmark dataset that contains 11 sequences (00-10) with images collected by car-mounted cameras and ground-truth trajectories provided by GPS. We used it for visual odometry experiment, with Sequences 00-08 for training and Sequences 09, 10 for testing. The images and ground truth are collected at 10 Hz.

KITTI Raw Dataset [15] contains both images (10 Hz) and high-frequency inertial data (100Hz). Since inertial data are only available in unsynced data packages, we selected the raw files with the corresponding to KITTI Odometry Dataset. Inertial data and images are manually synchronized according to their timestamps. We adopted the same data split mentioned above, discarding Sequence 03 as its raw data is unavailable.

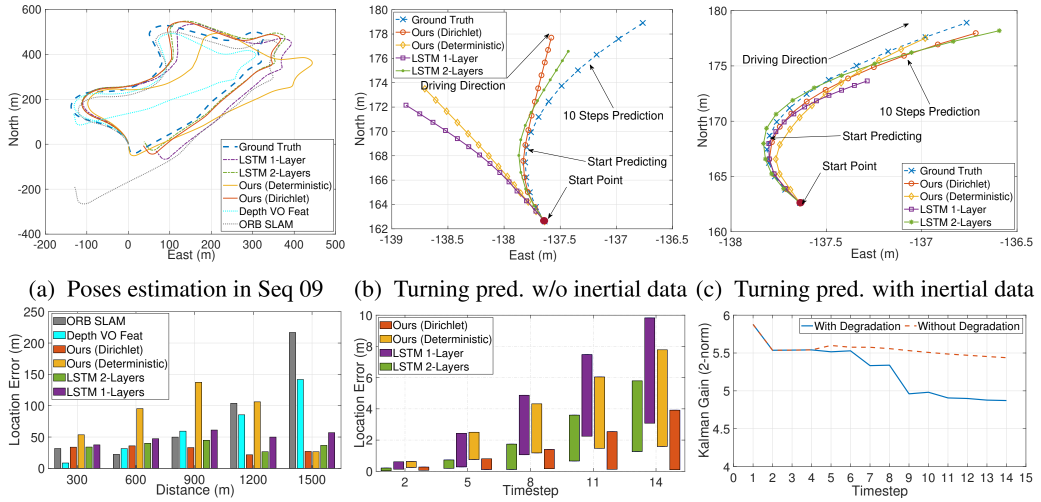
Pendulum Dataset [22] is a simulation of a dynamic torque-controlled pendulum. We follow the original dataset description and split it into different sets of 500 sequences for training, validating and testing respectively. Each sequence has 15 frames of 16×16 pixels and 1-dimensional control signal.

Experiment Setup: We implemented the proposed framework with Pytorch, and trained on a NVIDIA Titan X GPU. In VO and VIO experiments, we chose LSTMs (1-layer and 2-layers) as our baselines. The detailed framework structures are desribed in Supplementary Material. All the other modules for LSTMs including encoder and predictor, and the dimension of latent states (128) are kept the same as in our proposed models for a fair comparison. We also compare the trajectories from our model with two visual odometry/SLAM frameworks [1], i.e. Deep-VO-Feat [46] and ORB-SLAM [33]. In the pendulum control experiment, our model is compared with a state-of-the-art model Kalman VAE [13]. All of the models including the baselines are trained with the Adam optimizer with a batch-size of 32 and a learning rate of $1e^{-4}$.

3.2 Visual Egomotion Estimation

Our evaluation starts with a set of visual ego-motion (Visual Odometry) experiments for 6 DoF pose estimation and prediction. Table 1 reports on relative translation and orientation error [40], and summarises the comparison results in terms of root mean squared errors (RMSE). According to the findings from [40], the translational RMSE is sufficient enough to be a evaluation metric for visual odometry, as the rotational drifts show up as translational drifts when the camera is moving. Therefore, we mainly compare the different models based on their translational errors in Table 1, but only highlighted the rotational error when there is large discrepancy.

[1] The results are taken from https://github.com/Huangying-Zhan/Depth-VO-Feat



(a) Poses estimation in Seq 09 (b) Turning pred. w/o inertial data (c) Turning pred. with inertial data

(d) Localisation error of Seq. 09 (e) Error Bar of Straight line pred. (f) The 2-Norm of Kalman gain

Figure 2: The trajectories (a) and location errors (d) of Seq. 09 indicate that our models produce robust and accurate pose estimates in visual odometry. For future poses prediction without visual observations, our Dirichlet based model clearly outperform others when predicting the straight driving (e). In turning, the future poses are estimated in a tangent direction (b). With the aid of inertial data, our model can predict both the translation and rotation well (c). To interpret model, the Kalman gain reflects the measurements quantities, which decreases with the rising degree of data corruption (f).

Table 1: Visual Odometry on KITTI Odometry Dataset (Translation RMSE [0.01 m], Rotational RMSE [0.01 deg]).

	Pose Estimation	5 Steps Prediction	10 Steps Prediction
ORB-SLAM	17.7, 46.69	-	-
VO-Feat	10.99, 16.03	-	-
LSTM (1-layer)	5.81, 8.71	16.8, 50.7	23.4, 58.4
LSTM (2-layers)	5.76, 8.78	11.0, 36.3	17.7, 49.3
Our Model (Deterministic)	**5.60**, 8.86	10.8, 38.4	16.3, 47.1
Our Model (Dirichlet)	5.73, 8.72	**9.14**, 32.9	**13.5**, 49.2

VO Pose Estimation. In this experiment, a sequence of images are given to models to produce pose transformations. Both of our proposed models outperform the baselines of LSTMs and the largest gain is achieved by our deterministic model, as shown in Table 1. This experiment implies that the nonlinearities of VO systems are not lost despite the linear-like structures inside our models. Figure 2a illustrates the trajectories of Sequence 09 predicted by our models, showing competitive results compared with LSTMs, the classical ORB-SLAM [33] and a learning based model Depth-VO-Feat [46]. Sequence 09 is a difficult scenario, as the driving car experienced large movement in height. This degrades the accuracy of ORB-SLAM and Depth-VO-Feat, whereas our models can still provide robust and accurate results, which is clear to see the comparison of the absolute location errors in a travelled distance of 300m, 600m, 900, 1200m, 1500m in Figure 2d.

VO Pose Prediction. In another experiment, we fed models a sequence of 5 images for initialisation, and then let them predict the next 5 and 10 states without any further observations. As shown in Table 1, it is clear that our proposed Dirichlet based model performs best. This is because the resampled transition matrix from the Dirichlet distribution enables the dynamic model to be stable, and hence gives rise to long-term prediction in higher accuracy. As straight driving routes dominate driving behaviours for autonomous vehicles, we thus begin our performance report with them first. Figure 2e plots the error bar of predicted locations for straight driving, with the best and worst results among all tested segments to show the variance of the model predictions. In both cases, our proposed Dirichlet based model consistently outperforms other baselines by providing more accurate and robust location predictions. Figure 2b further shows the prediction performance in turning. As we can see, without timely observation, it is hard for deep neural networks to estimate accurate orientation changes, but they predict the future poses in a tangent direction. We will soon show how to integrate inertial information to aid the turning prediction in the Section 3.3.

Table 2: Visual-Inertial Navigation on KITTI Raw Dataset (Translation RMSE [0.01 m], Rotational RMSE [0.01 deg]).

	Poses Est.	Visual only Pred.	Inertial on Pred.	Pred. w/o Inputs
LSTM (1-layer)	6.24, 6.36	7.86, 150	23.7, 48.1	32.5, 118
LSTM (2-layers)	6.12, 6.99	6.90, 50.0	24.3, 26.4	21.2, 96.6
Ours (Deterministic)	**5.92**, 3.56	**6.27**, 27.3	**11.3**, 20.6	13.0, 39.2
Ours (Dirichlet)	6.03, 3.61	6.40, 32.3	12.3, 11.0	**12.3**, 40.9

3.3 Sensor Fusion for Visual-Inertial Navigation

How to effectively integrate and fuse two modalities to provide accurate and robust pose remains a challenging problem. In this experiment, we demonstrate that our proposed models can learn a compact state-space-model for sensor fusion from two modalities, i.e. visual and inertial data, and enable robust prediction under the circumstances with partial observations. Initially, a visual encoder and an inertial encoder extract m-dimensional visual features $\mathbf{a}_{\text{visual}} \in \mathbb{R}^m$ and n-dimensional inertial features $\mathbf{a}_{\text{inertial}} \in \mathbb{R}^n$ separately. These two features are then concatenated together as $\mathbf{a} = [\mathbf{a}_{\text{visual}}, \mathbf{a}_{\text{inertial}}] \in \mathbb{R}^{m+n}$. Notably, our emission matrix is defined as identity matrix $\mathbf{H} = \mathbf{I}_{m+n}$, when both modalities are available. If visual or inertial cues are absent, the emission matrix is changed to $\mathbf{H} = [\mathbf{I}_m, \mathbf{0}_{m \times n}]$ or $\mathbf{H} = [\mathbf{0}_{n \times m}, \mathbf{I}_n]$. The training and testing of the visual-inertial dynamic model follows the same procedures as in visual odometry.

VIO Pose Estimation. Table 2 reports the RMSE of the relative translation and orientation between the proposed models and LSTMs. Similar observations are found in this evaluation that our proposed models outperform 1-layer and 2-layers LSTMs when given both visual and inertial observations, with an especially high margin in orientation estimation. Although all models integrated inertial data to reduce the orientation errors of visual odometry, our proposed models excel at fusion.

VIO Pose Prediction. We also evaluate our models in scenarios of prediction with visual-only observations, inertial-only observations and no observations, in which all models are given a sequence of 5 images for initialisation and need to predict the next 5 poses. As can be seen in Table 2, our models greatly outperform the comparable deep learning approaches. Figure 2c plots the predicted trajectories with only inertial data, when no visual observation is given. Clearly, our models are capable of predicting future pose evolution accurately and robustly. We note that robust fusion is important for safe operation with missing sensor inputs e.g. for self-driving cars.

3.4 Pendulum Control

Last, we evaluate our models on the experiment of a dynamic torque controlled pendulum used in [22, 13] and we compare the performance of our model to two variants of Kalman VAE (KVAE) [13]. Following [22, 13], training, validation and test sets are formed by 500 sequences of 15 frames of 16×16 pixels. In the training phase, all models are learnt with the full length of 15 frames. However, in order to test the ability of prediction, we only feed the first 2 frames to these models and let them predict the following 13 frames with only control signals. In table 3, we see that our models achieve better performance compared with both deterministic and stochastic KVAE in terms of Binary Cross Entropy (BCE) and the accuracy of pixel agreement. This experiment implies that, beyond VO and VIO, our models are also able to reason about motion dynamics from visual data and use them to predict *future observations* when explicit control signals are given, e.g. the external forces placed on the pendulum.

Table 3: Pendulum Prediction

	BCE	Accuracy
KVAE (Det.)	2.71	0.922
KVAE (Stoc.)	2.72	0.921
Ours (Det.)	**2.66**	**0.923**
Ours (Dir.)	**2.66**	**0.923**

3.5 Towards Model Interpretability

We are now in a position to discuss model interpretability. Recall that in the update process (see Section 2.3), the Kalman gain is an adaptive weight that balances the observations and the model predictions. If there is high confidence in measurements, the Kalman gain will increase to selectively upweight measurement innovation and vice versa.

To understand model behaviours deeper, we deliberately fed our Dirichlet model with degraded images and use Kalman gain to capture the belief in measurements. This experiment generated 113 sequences with 15 frames of images from Sequence 09 in KITTI dataset. In each sequence, the images are corrupted with an increasing size of blanked blocks on the time steps 0-3 (no blocks), 4-5 (192 pixels * 192 pixels), 6-7 (192 pixels * 320 pixels), 8-9 (192 pixels * 480 pixels), 10-14 (192 pixels * 640 pixels). The values of Kalman gain are averaged over all the sequences on each timestep. We then test our model with this modified dataset in the same fashion as the VO experiment described in Section 3.2. The Frobenius norm (2-norm) of the Kalman gain matrix is calculated, and averaged across all sequences as an aggregated indicator of changes in Kalman gain matrix. Figure 2f shows that this indicator gradually decreases with increasing data corruption, comparing with the case without degradation. It implies that our model can adaptively place more trust on model predictions when observing low-quality data and signal to higher control layers that estimation is becoming more uncertain, allowing safer operation. It is critical to note that data corrupted in this way have never been seen in the training phase.

4 Related Work

State-Space-Models (SSMs) are a convenient and compact way to represent and predict system dynamics. In classical control engineering, system identification techniques are widely employed to build parametric SSMs [8]. In order to alleviate the effort of analytic description, data-driven methods, such as Gaussian Processes [23], Expectation–Maximization (EM) [16], or Gradient Descent [30], emerged as alternatives to identify nonlinear models. With advances in DNNs, deep SSMs have been recently studied to handle very complex nonlinearity. Specifically, [17, 20] used DNNs to extract effective features and feed them to a predefined physical model to improve filtering algorithms. Besides feature extraction, DNNs have also been used in re-parameterizing the transition matrix in SSMs from raw data [24, 14, 13, 22]. Unlike prior art, our work exploits recent findings on stable dynamical models [43] and use resampling to generate the transition matrix from Dirichlet distribution, whose concentration is learnt through RNNs. The specific Dirichlet distribution ensures the stability of dynamic systems, which is an important yet absent property of previous DNNs based SSMs.

Motion Estimation has been studied for decades and plays a central role in robotics and autonomous driving. Conventional visual odometry/SLAM methods rely on multiple-view geometry to estimate motion displacement between images [35, 7, 34, 10, 9, 12, 33]. Due to the huge availability and complementary property of inertial and visual sensors, effectively integrating these two sensor modalities has raised great attentions to give more robust and accurate inter-frame motion estimates [11]. A large portion of work in this direction is visual-inertial odometry, where filtering [27, 2] and nonlinear optimisation [26, 11, 36] are two mainstream model-based methods for sensor fusion. Meanwhile, recent studies also found that the methods based on state-of-the-art DNNs are able to provide competitive robustness and accuracy over some model-based methods. These DNN methods often use CNNs to learn useful geometry features from images for camera relocalization [47, 3, 46, 45, 42], and/or employ RNNs to model the temporary dependency in motion dynamics [6, 31, 44, 18]. Nevertheless, DNN based methods are hard to interpret and expect/modulate their behaviours [41, 37]. Our DynaNet aims to bridge the gap of performance and interpretability through a deeply coupled framework of model-based and DNN-based methods.

5 Conclusion

DynaNet, a neural Kalman dynamical model was introduced in this paper to learn temporal linear-like structure on latent states. Through deeply coupled DNNs and SSMs, DynaNet can scale to high-dimensional data as well as model very complex motion dynamics in real world. By using Kalman filter on feature space, DynaNet is able to reason about latent system states, allowing reliable inference and predictions even with missing observations. Furthermore, the transition matrix in our model is sampled from the Dirichlet distribution learned by a RNN, which ensures system stability. DynaNet is evaluated on a variety of challenging motion-estimation tasks, including single-modality estimation under data corruption, multiple sensor fusion under data absence, and learning with control signals. Experimental results demonstrate the superiority of our approach in accuracy, robustness and interpretability.

References

[1] Michael Bloesch, Jan Czarnowski, Ronald Clark, Stefan Leutenegger, and Andrew J. Davison. CodeSLAM — Learning a Compact, Optimisable Representation for Dense Visual SLAM. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[2] Michael Bloesch, Sammy Omari, Marco Hutter, and Roland Siegwart. Robust Visual Inertial Odometry Using a Direct EKF-Based Approach. In *IEEE International Conference on Intelligent Robots and Systems*, volume 2015-Decem, pages 298–304, 2015.

[3] Samarth Brahmbhatt, Jinwei Gu, Kihwan Kim, James Hays, and Jan Kautz. Geometry-Aware Learning of Maps for Camera Localization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2616–2625, 2018.

[4] Tayfun Çimen. *State-Dependent Riccati Equation (SDRE) Control: A survey*, volume 17. IFAC, 2008.

[5] Ronald Clark, Sen Wang, Andrew Markham, Niki Trigoni, and Hongkai Wen. VidLoc: A Deep Spatio-Temporal Model for 6-DoF Video-Clip Relocalization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[6] Ronald Clark, Sen Wang, Hongkai Wen, Andrew Markham, and Niki Trigoni. VINet: Visual-Inertial Odometry as a Sequence-to-Sequence Learning Problem. In *Association for the Advancement of Artificial Intelligence (AAAI)*, pages 3995–4001, 2017.

[7] Andrew J. Davison, Ian D. Reid, Nicholas D. Molton, and Olivier Stasse. MonoSLAM: Real-Time Single Camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1052–1067, 2007.

[8] D. G. Dudley. Dynamic system identification experiment design and data analysis. *Proceedings of the IEEE*, 67(7):1087–1087, July 1979.

[9] Jakob Engel, Thomas Schöps, and Daniel Cremers. LSD-SLAM: Large-Scale Direct Monocular SLAM. In *European Conference on Computer Vision (ECCV)*, 2014.

[10] Jakob Engel, Jurgen Sturm, and Daniel Cremers. Semi-Dense Visual Odometry for a Monocular Camera. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1449–1456, 2013.

[11] Christian Forster, Luca Carlone, Frank Dellaert, and Davide Scaramuzza. IMU Preintegration on Manifold for Efficient Visual-Inertial Maximum-a-Posteriori Estimation. In *Robotics: Science and Systems*, 2015.

[12] Christian Forster, Matia Pizzoli, and Davide Scaramuzza. SVO: Fast Semi-Direct Monocular Visual Odometry. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 15–22, 2014.

[13] Marco Fraccaro, Simon Kamronn, Ulrich Paquet, and Ole Winther. A Disentangled Recognition and Nonlinear Dynamics Model for Unsupervised Learning. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[14] Marco Fraccaro, Søren Kaae Sønderby, Ulrich Paquet, and Ole Winther. Sequential Neural Models with Stochastic Layers. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

[15] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision Meets Robotics: The KITTI Dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[16] Zoubin Ghahramani and Sam T Roweis. Learning Nonlinear Dynamical Systems Using an EM Algorithm. In *Advances in Neural Information Processing Systems (NIPS)*, volume 11, pages 431–437, 1999.

[17] Tuomas Haarnoja, Anurag Ajay, Sergey Levine, and Pieter Abbeel. Backprop KF: Learning Discriminative Deterministic State Estimators. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

[18] João F. Henriques and Andrea Vedaldi. MapNet: An Allocentric Spatial Memory for Mapping Environments. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8476–8484, 2018.

[19] Martin Jankowiak and Fritz Obermeyer. Pathwise Derivatives Beyond the Reparameterization Trick. In *International Conference on Machine Learning (ICML)*, 2018.

[20] Rico Jonschkowski, Divyam Rastogi, and Oliver Brock. Differentiable Particle Filters: End-to-End Learning with Algorithmic Priors. In *RSS*, 2018.

[21] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1):35, 1960.

[22] Maximilian Karl, Maximilian Soelch, Justin Bayer, and Patrick van der Smagt. Deep Variational Bayes Filters: Unsupervised Learning of State Space Models from Raw Data. In *International Conference on Learning Representations (ICLR)*, 2017.

[23] Juš Kocijan, Agathe Girard, Blaž Banko, and Roderick Murray-Smith. Dynamic Systems Identification with Gaussian Processes. *Mathematical and Computer Modelling of Dynamical Systems*, 11(4):411–424, 2005.

[24] Rahul G. Krishnan, Uri Shalit, and David Sontag. Structured Inference Networks for Nonlinear State Space Models. In *Association for the Advancement of Artificial Intelligence (AAAI)*, pages 1–21, 2017.

[25] Rainer Kümmerle, Giorgio Grisetti, Hauke Strasdat, Kurt Konolige, and Wolfram Burgard. g2o: A General Framework for Graph Optimization. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[26] Stefan Leutenegger, Simon Lynen, Michael Bosse, Roland Siegwart, and Paul Furgale. Keyframe-Based Visual–Inertial Odometry Using Nonlinear Optimization. *The International Journal of Robotics Research*, 34(3):314–334, 2015.

[27] Mingyang Li and Anastasios I. Mourikis. High-Precision, Consistent EKF-Based Visual-Inertial Odometry. *The International Journal of Robotics Research*, 32(6):690–711, 2013.

[28] Jun S. Liu and Rong Chen. Sequential monte carlo methods for dynamic systems. *Journal of the American Statistical Association*, 93(443):1032–1044, 1998.

[29] Piotr Mirowski, Matthew Koichi Grimes, Mateusz Malinowski, Karl Moritz Hermann, Keith Anderson, Denis Teplyashin, Karen Simonyan, Koray Kavukcuoglu, Andrew Zisserman, and Raia Hadsell. Learning to Navigate in Cities Without a Map. In *Advances in Neural Information Processing Systems (NIPS)*, 2018.

[30] Piotr Mirowski and Yann LeCun. Dynamical Factor Graphs for Time Series Modeling. In *ECML*, pages 1–18, 2009.

[31] Piotr Mirowski, Razvan Pascanu, Fabio Viola, Hubert Soyer, Andrew J. Ballard, Andrea Banino, Misha Denil, Ross Goroshin, Laurent Sifre, Koray Kavukcuoglu, Dharshan Kumaran, and Raia Hadsell. Learning to Navigate in Complex Environments. In *International Conference on Learning Representations (ICLR)*, 2017.

[32] S. Grewal Mohinder and P. Andrews Angus. Applications of Kalman Filtering in Aerospace 1960 to the Present. *IEEE Control Systems Magazine*, pages 69–78, 2010.

[33] Raul Mur-Artal, J.M.M Montiel, and Juan D. Tardos. ORB-SLAM : A Versatile and Accurate Monocular SLAM System. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015.

[34] Richard A Newcombe, Steven J Lovegrove, and Andrew J Davison. DTAM: Dense Tracking and Mapping in Real-Time. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2320–2327, 2011.

[35] D Nister, O Naroditsky, and J Bergen. Visual Odometry. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages I–652–I–659 Vol.1, 2004.

[36] Tong Qin, Peiliang Li, and Shaojie Shen. VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator. *IEEE Transactions on Robotics*, 34(4):1004–1020, 2018.

[37] Iyad Rahwan, Manuel Cebrian, Nick Obradovich, Josh Bongard, Jean François Bonnefon, Cynthia Breazeal, Jacob W. Crandall, Nicholas A. Christakis, Iain D. Couzin, Matthew O. Jackson, Nicholas R. Jennings, Ece Kamar, Isabel M. Kloumann, Hugo Larochelle, David Lazer, Richard McElreath, Alan Mislove, David C. Parkes, Alex 'Sandy' Pentland, Margaret E. Roberts, Azim Shariff, Joshua B. Tenenbaum, and Michael Wellman. Machine Behaviour. *Nature*, 568(7753):477–486, 2019.

[38] Syama Sundar Rangapuram, Matthias Seeger, Jan Gasthaus, Lorenzo Stella, Yuyang Wang, and Tim Januschowski. Deep State Space Models for Time Series Forecasting. *Advances in Neural Information Processing Systems*, pages 7795–7804, 2018.

[39] John Schulman, Nicolas Heess, Theophane Weber, and Pieter Abbeel. Gradient Estimation Using Stochastic Computation Graphs. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1–13, 2015.

[40] Jrgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A Benchmark for the Evaluation of RGB-D SLAM Systems. In *IEEE International Conference on Intelligent Robots and Systems, (IROS)*, pages 573–580, 2012.

[41] Niko Sünderhauf, Oliver Brock, Walter Scheirer, Raia Hadsell, Dieter Fox, Jürgen Leitner, Ben Upcroft, Pieter Abbeel, Wolfram Burgard, Michael Milford, and Peter Corke. The Limits and Potentials of Deep Learning for Robotics. *International Journal of Robotics Research*, 37(4-5):405–420, 2018.

[42] Chengzhou Tang and Ping Tan. BA-Net: Dense Bundle Adjustment Networks. In *International Conference on Learning Representations (ICLR)*, 2019.

[43] Jonas Umlauft and Sandra Hirche. Learning Stable Stochastic Nonlinear Dynamical Systems. In *International Conference on Machine Learning (ICML)*, pages 3502—-3510, 2017.

[44] Sen Wang, Ronald Clark, Hongkai Wen, and Niki Trigoni. DeepVO : Towards End-to-End Visual Odometry with Deep Recurrent Convolutional Neural Networks. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2017.

[45] Zhichao Yin and Jianping Shi. GeoNet: Unsupervised Learning of Dense Depth, Optical Flow and Camera Pose. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[46] Huangying Zhan, Ravi Garg, Chamara Saroj Weerasekera, Kejie Li, Harsh Agarwal, and Ian Reid. Unsupervised Learning of Monocular Depth Estimation and Visual Odometry with Deep Feature Reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 340–349, 2018.

[47] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G. Lowe. Unsupervised Learning of Depth and Ego-Motion from Video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.