

# Dynamical Variational Autoencoders and KalmanNet: New Approaches to Robust High-Precision Navigation

Dan Shen<sup>1</sup>, Yuexin Ma<sup>2</sup>, Gelu Liu<sup>2</sup>, Jiaocheng Hu<sup>2</sup>, Qizhen Weng<sup>2</sup>, Xiangwei Zhu<sup>2\*</sup>

<sup>1</sup> School of Systems Science and Engineering, Sun Yat-sen University, 510006, Guangzhou, China - shend7@mail2.sysu.edu.cn

<sup>2</sup> School of Electronics and Communication Engineering, Sun Yat-sen University, 528406, Shenzhen, China - (mayx58, zhuxw666@mail.sysu.edu.cn, (liuglu, hujch23, wengqz3@mail2.sysu.edu.cn

**KEY WORDS:** State Space Models, Dynamical Variational Autoencoders, Dynamic Bayesian Networks, Kalman Filter, KalmanNet.

## ABSTRACT:

Kalman filters, recognized as a traditional and effective inference algorithm based on state space models (SSM), have been extensively applied in the fields of navigation and mapping. However, their performance will degrade when facing model assumption mismatches, such as non-linear dynamics and non-Gaussian correlated noises. The model-based deep learning methods overcome these mismatches by combining the domain knowledge of the model-based methods and the expressiveness of the data-driven deep learning methods, and thus can provide a promising solution for addressing high-dimensional and nonlinear challenges. This paper presents a succinct overview of the principles, inference model, and training methodology employed in model-based deep learning methods, with particular focus on the KalmanNet and the Dynamical Variational Autoencoder (DVAE). Furthermore, it implements KalmanNet on robust and high-precision navigation and positioning problem. The experimental results substantiate the feasibility of achieving navigation and positioning accuracy comparable to that of the Extended Kalman Filter (EKF), while simultaneously exhibiting enhanced robustness, albeit at the cost of some computational overhead.

## 1. INTRODUCTION

The Kalman filter is extensively employed in aerospace, navigation, and precise guidance domains, prized for its capability to real-time fuse multi-sensor information. Among them, the filter based on the combination of global navigation satellite system (GNSS) and inertial navigation system (INS), holds particularly widespread application (Proletary, 2019, Seleznev et al., 2019). In the context of INS correction, different estimation algorithms are typically employed. For instance, a nonlinear Kalman filter is often utilized (Zheng et al., 2018, Simon, 2006, Julier, 1997). In order to establish a dependable and comprehensive model, concepts from the federated Kalman filter and model construction methodologies were incorporated (Yang et al., 2020, Antonito R, 2005, Kondo T, 1998). Built upon state space models (SSM), this variant of the Kalman filter relies on conventional signal processing techniques, which heavily depend on manually designed simple mathematical models derived from domain expertise and assume Gaussian characteristics in the random models (Shlezinger, 2023). While the Kalman filter offers advantages like a compact footprint, minimal delay, and low power consumption, it encounters challenges when dealing with nonlinear state models and non-Gaussian random models (Yan et al., 2022, Das et al., 2015, Saha et al., 2022).

In recent years, parameterized deep neural network models have seen widespread and successful applications in various domains, including computer vision (LeCun et al., 2015, He et al., 2015), and complex games like Go (Silver et al., 2017) and Starcraft (Vinyl's et al., 2019). This potent nonlinear function approximator has gradually emerged as a novel and more efficient approach to signal processing. Data-driven methods like DNNs offer two advantages over model-based approaches. Firstly, purely data-driven techniques operate without relying

on analytical approximations, allowing them to function even when analytical models are unknown. Secondly, for complex systems, data-driven algorithms can extract the necessary features for inference from observed data (Bengio, 2009). DNN-based data-driven methods leverage autoregressive exploration within extensive data containing both inherent knowledge and noise, extracting hidden structural and model information. While these methods are powerful tools, they also exhibit challenges such as heavy data dependency, significant computational requirements, convergence difficulties, and the opaque nature of the model. Consequently, deep learning has not yet attained the interpretability, flexibility, versatility, and reliability that model-based methods provide (Chen and Ran, 2019).

SSM and deep learning offer complementary strengths. By leveraging conditional independence encoded in the probabilistic graphical representation and inference techniques along with partial domain knowledge, it's possible to construct modularly designed deep neural networks to replace unknown nonlinear function models and high-dimensional, highly correlated models. This holds the potential to synergize the strengths of both approaches in signal processing, leading to more effective processing methods (Shlezinger, 2023, Girin et al., 2020). This article is mainly informed by the groundbreaking contributions of Nir Shlezinger et al. (Shlezinger, 2023) and Laurent Girin et al. (Girin et al., 2020). It offers a succinct introduction to the Model-Based Deep Learning approach method and the Dynamical Variational Autoencoder, analyzing their fundamental conceptual frameworks and applicable domains. Furthermore, it seeks to integrate this efficient and novel modern signal processing approach into the domain of navigation.

---

\* Corresponding author

## 2. DISCUSSIONS

Traditional outdoor navigation primarily relies on GNSS, RTK (Real-time kinematic) and INS, utilizing Kalman filters to combine the positional observations from GNSS or RTK with the state updates from INS. For handling nonlinear models, approaches often involve linearized Extended Kalman Filter (EKF) models or methods like the federated Kalman filter. In the context of indoor navigation or autonomous driving scenarios, incorporating multi-modal information such as RGB vision, depth maps, and even sound for data feature extraction and fusion represents a wise, feasible, and promising technological approach with broad prospects. However, combining the richness of high-dimensional sensor data like vision with the integration of low-dimensional sensor data such as INS and GNSS naturally presents two primary challenges: first, dealing with the nonlinearity of function modeling and non-Gaussian characteristics of random modeling; and second, efficiently extracting features to map high-dimensional, information-rich data to low-dimensional target information. Model-Based Deep Learning and Dynamical Variational Autoencoders can each propose solutions for the two challenges mentioned above.

### 2.1 Model-Based Deep Learning

Model-Based Deep Learning (MB-DL) is an approach that combines prescribed data models such as state space model (SSM) and deep neural networks to compensate for mismatches in the data models due to complex dynamics and intricate random distributions that are challenging to analytically describe. Through parameterized networks, this approach embeds these learnable but previously unknown aspects into the DNN via data-driven "learning". This integration ensures the fusion of domain knowledge with data-driven DNN techniques. By merging the strengths of DNNs and SSMs, this approach equips the MB-DL method with enhanced precision, efficiency, and robustness. In this context, the term "efficiency" might be slightly ambiguous. From our perspective, "efficiency" refers to the synergy achieved by blending the strengths of DNNs and SSMs in the MB-DL method. This integration allows for a more accurate and robust approach to handling complex data relationships.

Figure 1 provides a comparison between MB-DL and data-driven DNNs. From the diagram, it is evident that the so-called MB-DL combines the strengths of SSM and DNN, and it is divided into two main categories based on domain knowledge or data-driven foundation: model-aided networks and DNN-aided inference. Figure 2 illustrates Nir Shlezinger's comprehensive categorization of MB-DL, along with several prominent methods within each category (Shlezinger, 2023).

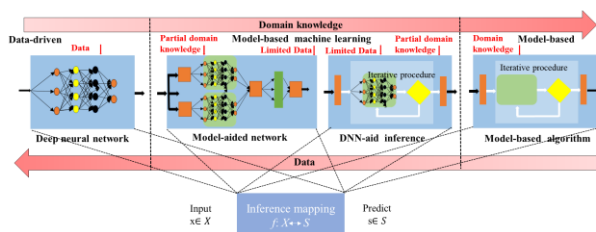


Figure 1. Model-based versus Data-driven approaches (Shlezinger, 2023)

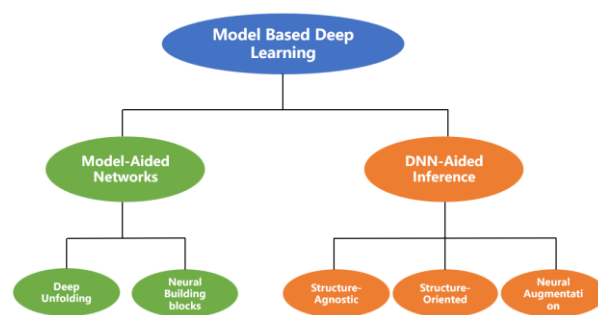


Figure 2. Taxonomy of model-based deep learning approaches (Shlezinger, 2023).

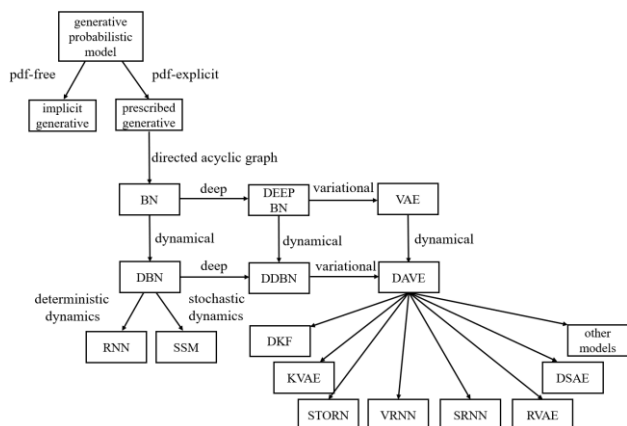
As discussed above, the MB-DL method essentially leverages pre-trained parameterized DNN models with richer high-value data to compensate for the limitations of existing SSM methods, such as Kalman filtering. KalmanNet is a real-time state estimator that integrates deep learning with the Kalman filtering model, serving as an effective solution to address nonlinearity and model mismatch in dynamic environments. By synergizing the model-based Extended Kalman Filter (EKF) algorithm with recurrent neural networks from the realm of deep learning, KalmanNet excels in providing accurate and reliable estimates of the underlying state variables, even in scenarios where noise statistical properties remain unknown. This innovative approach not only enhances state estimation precision in the presence of uncertainties, but also showcases the potential for harmonizing traditional estimation techniques with modern machine learning paradigms (Revach et al., 2022).

However, its essence remains rooted in the SSM model. In comparison to traditional domain-knowledge-based methods, the MB-DL approach introduces an additional "learning" step, thus compensating for the absence of prior domain knowledge. This implies that it has two drawbacks: (1) It requires a substantial amount of high-value data for pre-training the DNN, and it's well-known that such data often comes at a higher cost. (2) The intricate architecture of deep neural networks can make it challenging to fully grasp how the model arrives at its decisions. This limitation can hinder transparency and clarity, especially in applications where comprehending the decision-making process is crucial. In simpler terms, due to the complex nature of the neural network's inner workings, it becomes harder to explain the reasoning behind its choices. This can pose a problem in fields where understanding the 'why' is essential.

The MB-DL method inherently assumes that domain knowledge and intricate unknown random distributions can be effectively approximated by parameterized DNN networks. Although Nir Shlezinger does not explicitly mention this, the premise remains unavoidable. However, we hold an optimistic view on this matter. We firmly believe in the potent expressive capabilities of DNNs, even though, in certain applications, the goal of DNN approximation might not strictly be categorized as a functional one. Additionally, it's noteworthy that training the DNN in MB-DL is predominantly based on supervised learning methods. As of now, more effective unsupervised learning, imitation learning, and reinforcement learning techniques have not been seamlessly integrated into this framework. Utilizing supervised learning for training necessitates explicit inputs and outputs for the network, as well as high-quality data, all of which pose limitations on the applicability of MB-DL. Nonetheless, MB-DL remains an impressive achievement. It enables inexpensive sensors to achieve accuracy levels comparable to expensive sensors through the effective utilization of substantial prior information.

## 2.2 Dynamical Variational Autoencoders

As another member of the model-based deep learning family, the Dynamical Variational Autoencoder (DVAE) can be viewed as the combination of the directed probabilistic graphical model and the variational autoencoder training methodology. Compared with KalmanNet, the DVAE is more closely related to the probabilistic generative model and can be employed as an alternative method to deal with the model mismatching problem.



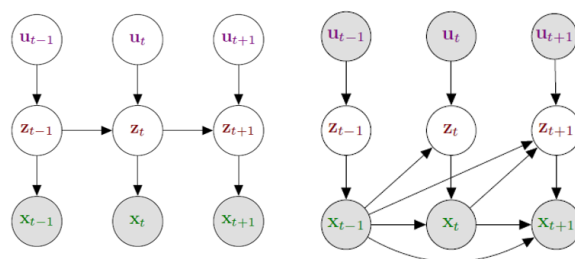
**Figure 3.** Autoencoders (AE), Variational Autoencoders (VAE) and Dynamical Variational Autoencoders (DVAE) in the taxonomy of generative models (Girin et al., 2020).

To introduce DVAE, it's helpful to first understand Autoencoders (AE) and Variational Autoencoders (VAE). Autoencoders (AE) are neural network architectures designed for unsupervised learning. They consist of an encoder and a decoder. The encoder compresses the input data into a lower-dimensional representation called the "latent space," and the decoder tries to reconstruct the original input data from this latent representation. The AE's primary aim is to learn an efficient data representation. Variational Autoencoders (VAE) are an extension of AE that brings in probabilistic modeling. VAEs introduce a probabilistic interpretation to the latent space, allowing the model to generate new data samples by sampling from the latent space distribution. The VAE framework employs a loss function that encourages the learned latent space to follow a specific probability distribution, typically a Gaussian distribution. This results in a continuous and smooth latent space, making it useful for data generation and interpolation. Dynamical Variational Autoencoders (DVAE) are a further development that introduces the concept of temporal dynamics. They aim to capture the temporal dependencies present in sequential data. By incorporating recurrent neural networks (RNNs) into the VAE framework following the conditional dependencies implied by probabilistic graphical models, DVAEs can model time-evolution of the latent representation and more diverse dependencies between the external control, hidden states, and observational variables. This makes DVAEs particularly suited for tasks involving complex time-series data or sequential data, such as speech data and video frames over time. Figure 3 illustrates the placement of Autoencoders (AE), Variational Autoencoders (VAE), and Dynamical Variational Autoencoders (DVAE) within the framework of probabilistic generative models (Girin et al., 2020).

In essence, the goal of an Autoencoder (AE) is to map high-dimensional, sparse, and structured data to low-dimensional feature vectors using an encoding-decoding structure. This

process avoids supervised training and directly maps the data to hidden states. Variational Autoencoder (VAE) extends this by introducing probabilistic encoding which ensures better generalization ability of the model. Dynamical Variational Autoencoder (DVAE) further extends VAE by introducing probabilistic dependencies across time. This is imposed by the probabilistic graphical structure of the model and can extend the first-order Markovian assumption in SSM to more complex conditional dependencies, as shown in Figure 4.

In Figure 4, the classical SSM (left) assumes that the evolution of hidden variables  $z$  obeys the first-order Markov property, and observational variables  $x$  provide no effective information for inferring  $z$ . In DVAE, however, we can model more flexible probabilistic dependencies between variables and hence higher-order Markov property, which will enhance the representational capability of the model. For instance, in the right subfigure of Figure 4, direct dependencies between hidden variables are replaced by indirect dependencies via observational variables, which additionally obey the second-order Markov property.



**Figure 4.** Graphical illustration of SSM (left) and an instance of DVAE (right) (Girin et al., 2020).

These complex probabilistic dependencies can be implemented by different architectures of DNNs, particularly Recurrent Neural Networks (RNNs) which can memorize the accumulated past information and is theoretical guaranteed to approximate any dynamical system. The Deep Kalman Filter (DKF) (Krishnan, 2021) is an instance of a DVAE model. In DKF, the linear state transition and observation models of the Kalman Filter are replaced by DNNs to model more intricate probabilistic dependencies across time. These DNNs can capture complex and nonlinear dynamical relationships and factors of variations in the data. DKF combines the recursive filtering and smoothing operations of Kalman Filters with the powerful representation learning capacity of DNNs, making it suitable for handling complex, nonlinear, and high-dimensional data. It effectively learns the system's dynamics from data, thus providing more accurate state estimates and predictions compared to traditional Kalman Filters in scenarios with nonlinearities and non-Gaussian correlated noise.

The primary challenge with DVAE is its difficulty in training and ensuring the correctness and effectiveness of the encoded information within DNNs. In practice, DVAE represents a typical deep neural network model. Despite its utilization of the more expressive probabilistic graphical model, it lacks the interpretability advantages of traditional SSM models and the MB-DL framework. However, to introduce images and true semantics (beyond instance segmentation) into navigation, achieving genuine semantics (which is inherently one-to-many and not strictly a function) – for instance, considering "cat" as an abstract category encompassing various depictions, even cartoons – requires a deeper network model based on probabilistic graphical dependencies. This approach holds

greater promise for achieving biologically inspired navigation resembling heuristic brain processes.

### 3. EXPERIMENTS AND RESULTS

In this section, we will employ a simulation environment to conduct a brief efficacy assessment of the MB-DL (KalmanNet) approach.

(a) In our first experimental study, the constant acceleration model is used to compare KalmanNet to the MB KF which is known to minimize the MSE in such a setup.

(b) We next evaluate the filtering performance of KalmanNet within the context of the Lorenz attractor model and to compare it against the Extended Kalman Filter (EKF) methodology.

#### 3.1 Experimental Setting

In our experimental setup, we introduce two distinct technical terms to describe the nonlinear model: "full information" and "partial information" (Revach et al., 2022).

*Full information:* KalmanNet operates with complete knowledge of the dynamical and observational functions  $f(\cdot)$  and  $h(\cdot)$ , but no access to the noise covariance matrix  $Q$  and  $R$ , whereas its model-based counterpart operates with exact knowledge of the matrices.

*Partial information:* The operation of KalmanNet and its model-based counterparts involves a certain degree of model mismatch. This will be elaborated upon in detail in the subsequent experiments.

The performance evaluation metric utilized is Mean Squared Error (MSE) in [dB], with the Adam optimizer employed.

#### 3.2 Result

##### 3.2.1 Linear State Space Model

In order to demonstrate the applicability of KalmanNet to various linear systems, we set the dimension of the system to  $2 \times 2$ , the sequence length of the training set is 20, but the test set has trajectories of different lengths {50, 100, 200}. Figure 5, Figure 6, and Figure 7 show the tracking effect of KalmanNet and Kalman Filter on position when  $R=0.01$ .

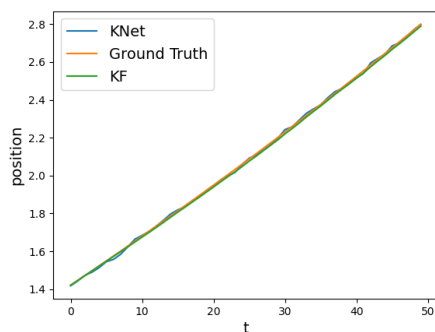


Figure 5. T=50, R=0.01

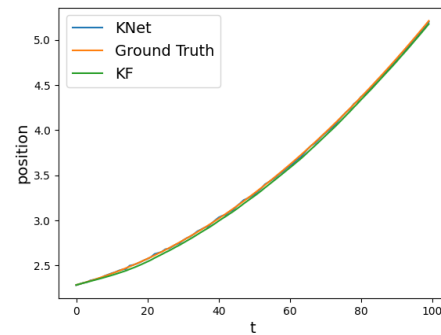


Figure 6. T=100, R=0.01

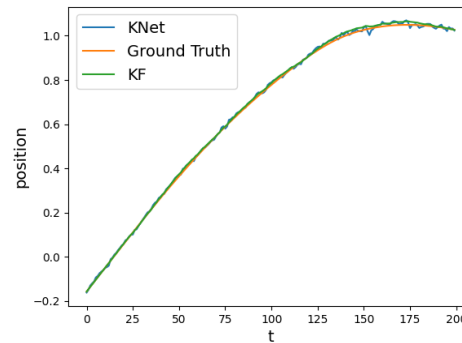


Figure 7. T=200, R=0.01

We compare the filtering performance of KalmanNet and Kalman Filter from three aspects. MSE LOSS most directly reflects the quality of filtering, the standard deviation (STD) reflects the stability of filtering, and finally, the inference time reflects the timeliness of filtering. Table 1, Table 2, and Table 3 provides a comprehensive presentation of the test results. we can clearly observe that KalmanNet achieves the MMSE of the MB KF. And KalmanNet shows better stability and shorter inference time.

|      | MSE LOSS        | STD           | Inference Time |
|------|-----------------|---------------|----------------|
| KF   | -34.7239        | 2.6657        | 1.6261         |
| KNet | <b>-41.2537</b> | <b>0.9151</b> | <b>0.1432</b>  |

Table 1. T=50, R=0.01

|      | MSE LOSS        | STD           | Inference Time |
|------|-----------------|---------------|----------------|
| KF   | -33.7196        | 2.5504        | 1.5860         |
| KNet | <b>-40.3052</b> | <b>0.9678</b> | <b>0.1555</b>  |

Table 2. T=100, R=0.01

|      | MSE LOSS        | STD           | Inference Time |
|------|-----------------|---------------|----------------|
| KF   | -33.2109        | 2.8877        | 1.5588         |
| KNet | <b>-39.0541</b> | <b>1.8400</b> | <b>0.4156</b>  |

Table 3. T=200, R=0.01

##### 3.2.2 Non-Linear Model

The Lorenz attractor represents a three-dimensional chaotic solution within the continuous-time Lorenz system of ordinary differential equations. This synthetically generated system serves to illustrate the task of dynamically tracking a profoundly nonlinear trajectory in an online fashion. Additionally, it addresses a practical challenge encountered in the real world, involving the management of disparities arising from the discretization of continuous-time signals into discrete-time samples.

**Noisy state observations:** The observed outcomes are versions of the true state corrupted by noise. With a training sequence length of  $T=100$  and a testing sequence length of  $T=1000$ , Table 4 and Table 5 provides a comprehensive presentation of the test results. Under the condition of training with relatively shorter sequence lengths, KalmanNet achieves comparable MSE performance to the Extended Kalman Filter (EKF), but with longer filtering time

| $1/r^2$ [dB] | 0               | 10              | 20              | 30              | 40              |
|--------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| EKF          | <b>-10.3642</b> | <b>-20.4102</b> | <b>-30.3726</b> | <b>-40.2716</b> | <b>-49.9020</b> |
| KNet         | -9.3863         | -18.9256        | -29.3419        | -38.1881        | -45.1282        |

**Table 4.** MSE [dB]-Noisy state observations

| $1/r^2$ [dB] | 0             | 10            | 20            | 30            | 40            |
|--------------|---------------|---------------|---------------|---------------|---------------|
| EKF          | <b>170.34</b> | <b>169.16</b> | <b>170.37</b> | <b>170.55</b> | <b>174.10</b> |
| KNet         | 220.92        | 231.76        | 233.46        | 224.95        | 227.02        |

**Table 5.** Inference Time -Noisy state observations

**Noisy non-linear observations:** The observations are generated through a nonlinear function of the current state, with  $h$  set to represent the transformation from Cartesian coordinates to spherical coordinates. From the results reported in Table 6 we observe that the Extended Kalman Filter (EKF) experiences a complete degradation in filtering performance, whereas KalmanNet is capable of maintaining relatively favorable performance.

| $1/r^2$ [dB] | -10            | 0             | 10             | 20              | 30              |
|--------------|----------------|---------------|----------------|-----------------|-----------------|
| EKF          | nan            | nan           | nan            | 31.7123         | 31.5218         |
| KNet         | <b>13.2830</b> | <b>6.0234</b> | <b>-4.7157</b> | <b>-10.6565</b> | <b>-14.4138</b> |

**Table 6.** MSE [dB]-noisy non-linear observations

**State-evolution mismatch:** The data generation involves extending the Taylor expansion series to  $J=5$ , whereas both KalmanNet and the Model-Based (MB) algorithm operate with coarse approximations ( $J=2$ ). From the results reported in Table 7 and Table 8 we observe that in the presence of state-evolution mismatch, KNet effectively mitigates this discrepancy, exhibiting filtering performance that remains closely aligned with that achieved under full information conditions. Similarly, the Extended Kalman Filter (EKF) demonstrates the ability to overcome such mismatch when  $1/r^2$  is within the range of 10-20. However, the ability of EKF to address the mismatch significantly deteriorates when  $1/r^2$  exceeds the range of 30-40. Kalman filter once again showed better timeliness than KalmanNet.

| $1/r^2$ [dB] | 10              | 20              | 30              | 40              |
|--------------|-----------------|-----------------|-----------------|-----------------|
| EKF          | <b>-19.5333</b> | -25.4125        | -26.8325        | -28.6127        |
| KNet         | -19.0504        | <b>-26.4003</b> | <b>-34.4143</b> | <b>-41.2624</b> |

**Table 7.** MSE [dB]-State-evolution mismatch

| $1/r^2$ [dB] | 10            | 20            | 30            | 40            |
|--------------|---------------|---------------|---------------|---------------|
| EKF          | <b>159.10</b> | <b>162.19</b> | <b>162.08</b> | <b>162.99</b> |
| KNet         | 199.25        | 192.90        | 195.53        | 201.83        |

**Table 8.** Inference Time-State-evolution mismatch

**State-observation rotation mismatch:** Data generated by utilizing the identity matrix with only a  $1^\circ$  rotation ( $\theta = 1^\circ$ ) is employed to simulate the mismatch present in the observation model. The results reported in Table 9 and Table 10 clearly demonstrate that under the condition of state-observation rotation mismatch, KalmanNet effectively overcomes this

mismatch within the range of  $1/r^2$  from 0 to 20, exhibiting filtering performance that remains closely aligned with that achieved under full information conditions. However, as  $1/r^2$  reaches 30, there is a moderate decrease in its ability to handle this mismatch. Similarly, the Extended Kalman Filter (EKF) demonstrates the ability to address this mismatch within the range of  $1/r^2$  from 0 to 10. Nevertheless, as  $1/r^2$  ranges from 20 to 30, there is a substantial decline in its capability to mitigate the mismatch.

| $1/r^2$ [dB] | 0              | 10              | 20              | 30              |
|--------------|----------------|-----------------|-----------------|-----------------|
| EKF          | -9.5291        | -15.3676        | -16.8803        | -16.8965        |
| KNet         | <b>-9.5543</b> | <b>-19.2013</b> | <b>-28.1989</b> | <b>-33.2600</b> |

**Table 9.** MSE [dB]-State-observation rotation mismatch

| $1/r^2$ [dB] | 0            | 10           | 20           | 30           |
|--------------|--------------|--------------|--------------|--------------|
| EKF          | <b>84.09</b> | <b>84.33</b> | <b>85.62</b> | <b>85.38</b> |
| KNet         | 110.28       | 113.31       | 111.13       | 110.85       |

**Table 10.** Inference Time -State-observation rotation mismatch

#### 4. CONCLUSION

This paper introduces two novel modern data processing methods: KalmanNet and Dynamical Variational Autoencoder (DVAE), which can be unified in the framework of Model-based Deep Learning (MB-DL). It explores the principles, advantages, applicable domains, and how to effectively apply them to navigation tasks. MB-DL is essentially a information processing method that combines the advantage of domain knowledge implied by assumed models and the expressiveness of the DNNs. It exhibits higher interpretability and efficiency while maintaining a certain level of robustness compared to pure data-driven DNNs. It is particularly effective in traditional navigation filtering algorithms, showcasing its prowess. However, it demands more expensive data support, such as high-precision "truth" obtained through more accurate sensors or measurement methods. With access to this data, MB-DL can achieve results closely approximating those obtained from costlier measurement techniques. In our experiments, the MB-DL (KalmanNet) method showed comparable performance to Kalman Filter on linear models, but with shorter inference time. In nonlinear models, it outperformed EKF in the face of model mismatch, demonstrating the benefit of incorporating data-driven RNN into the model-based filtering method. However, MB-DL may struggle with feature extraction and processing high-dimensional semantic information. It can be effectively utilized after an efficient feature extractor. Interestingly, DVAE conveniently addresses this limitation. The application of DVAE extends beyond just positioning and trajectory estimation; it represents a promising approach for achieving intelligent navigation. It holds a natural advantage over MB-DL in processing high-dimensional semantic information. By combining probabilistic graphical representation and the VAE methodology for training DNNs, DVAEs utilize neural networks to describe prior information that is challenging to represent mathematically. This approach offers greater flexibility and effectiveness. However, it inherits the common shortcomings of DNNs: difficulty in training, lack of interpretability, and challenges in generalization. However, the integration and application of both MB-DL and DVAE methods could potentially bring limitless new possibilities to intelligent navigation tasks.

## ACKNOWLEDGEMENTS

This work was supported by National Natural Science Foundation of China (Grant No.61973328); Shenzhen Science and Technology Program (Grant No. GXWD 20201231165807008, 20200830225317001); Shenzhen Science and Technology Plan Project (Grant No. ZDSYS20210623091807023).

## REFERENCES

- Bengio, Y., 2009. Learning deep architectures for AI. *Foundations and trends® in Machine Learning*, 2(1), 1-127.
- Chen, J., & Ran, X., 2019. Deep learning with edge computing: A review. *Proceedings of the IEEE*, 107(8), 1655-1674.
- Das, J., Cross, G., Qu, C., Makineni, A., Tokekar, P., Mulgaonkar, Y., & Kumar, V., 2015. Devices, systems, and methods for automated monitoring enabling precision agriculture. In *2015 IEEE international conference on automation science and engineering (CASE)*, 462-469.
- Fantinutto R, Guglieri G, Quagliotti FB., 2005. Flight control system design and optimization with a genetic algorithm. *Aerospace Science and Technology*, 9(1), 73–80.
- Girin, L., Leglaive, S., Bie, X., Diard, J., Hueber, T., & Alameda-Pineda, X., 2020. Dynamical variational autoencoders: A comprehensive review. *Foundations and trends® in Machine Learning*.
- He, K., Zhang, X., Ren, S., & Sun, J., 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, 1026-1034.
- Julier SJ, Uhlmann JK., 1997. A new extension of the Kalman filter to nonlinear systems. In *Signal processing, sensor fusion, and target recognition VI*, Vol. 3068, 182-193.
- Kondo T., 1998. GMDH neural network algorithm using the heuristic self-organization method and its application to the pattern identification problem. In: *Proceedings of the 37th SICE Annual Conference. International Session Papers*, 1143–1148.
- Krishnan, R. G., Shalit, U., & Sontag, D., 2015. Deep kalman filters. arXiv preprint arXiv:1511.05121.
- LeCun, Y., Bengio, Y., & Hinton, G., 2015. Deep learning. *nature*, 521(7553), 436-444.
- Proletary AV, Neomycin KA, Seleznev MS., 2019. Method for improving accuracy of INS using scalar parametric identification. In: *International Russian Automation Conference*, 1–4.
- Revach, G., Shlezinger, N., Ni, X., Escoriza, A. L., Van Sloun, R. J., & Eldar, Y. C., 2022. KalmanNet: Neural network aided Kalman filtering for partially known dynamics. *IEEE Transactions on Signal Processing*, 70, 1532-1547.
- Saha, S. S., Sandha, S. S., Garcia, L. A., & Srivastava, M., 2022. Tinyodom: Hardware-aware efficient neural inertial navigation. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 6(2), 1-32.
- Seleznev M. S., Proletary A. V., Neomycin K. A., Lifei Z., 2019. Modification of the federated Kalman filter using the observability degree criterion of state variables. In: *26th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS)*, 1–3.
- Shlezinger, N., Whang, J., Eldar, Y. C., & Dimakis, A. G., 2023. Model-based deep learning. *Proceedings of the IEEE*.
- Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., ... & Hassabis, D., 2017. Mastering the game of go without human knowledge. *Nature*, 550(7676), 354-359.
- Vinyals, O., Babuschkin, I., Czarnecki, W. M., Mathieu, M., Dudzik, A., Chung, J., ... & Silver, D., 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature*, 575(7782), 350-354.
- Yan, Y., Zhang, B., Zhou, J., Zhang, Y., & Liu, X. A., 2022. Real-Time localization and mapping utilizing multi-sensor fusion and visual-IMU-wheel odometry for agricultural robots in unstructured, dynamic and GPS-denied greenhouse environments. *Agronomy*, 12(8), 1740.
- Yang Y, Liu X, Zhang W, Liu X, Guo Y., 2020. A nonlinear double model for multisensor-integrated navigation using the federated EKF algorithm for small UAVs. *Sensors*, 20(10), 2974.
- Zheng B, Fu P, Li B, Yuan X., 2018. A robust adaptive unscented Kalman filter for nonlinear estimation with uncertain noise covariance. *Sensors*, 18(3), 808.