A DEEP-LEARNING-AUGMENTED KALMAN FILTER FOR HIGH-PERFORMANCE INTRACORTICAL DECODING

Luis H. Cubillos[1], Guy Revach[3], Joseph Costello[1], Hisham Temmar[1], Matthew J. Mender[1], Xiaoyong Ni[3], Madison Kelberman[1], Dylan Wallace[1], Matthew S. Willsey[2], Ruud J.G. van Sloun[4], Nir Shlezinger[5], Parag G. Patil[2], **Cynthia A. Chestek[1]**

[1]Cortical Neural Prosthetics Lab, Departments of Robotics and Biomedical Engineering, University of Michigan; [2]Department of Neurosurgery, University of Michigan; [3]Department of Information Technology and Electrical Engineering, ETH Zürich; [4]Department of Electrical Engineering, Eindhoven University of Technology; [5]School of Electrical and Computer Engineering, Ben-Gurion University

MICHIGAN ENGINEERING ROBOTICS
UNIVERSITY OF MICHIGAN

Scan here for more info

MOTIVATION

Brain-machine interfaces (BMIs) allow patients with spinal cord injury to recover some independence by controlling external devices using the signals from their brains.

Decoding models, which take the brain signals and output a control signal, have flourished in the past decades, with **deep learning models** showing good performance. However, with **physical systems** (e.g., robotic arm), when the patients' safety is critical, **linear models are still the standard**.

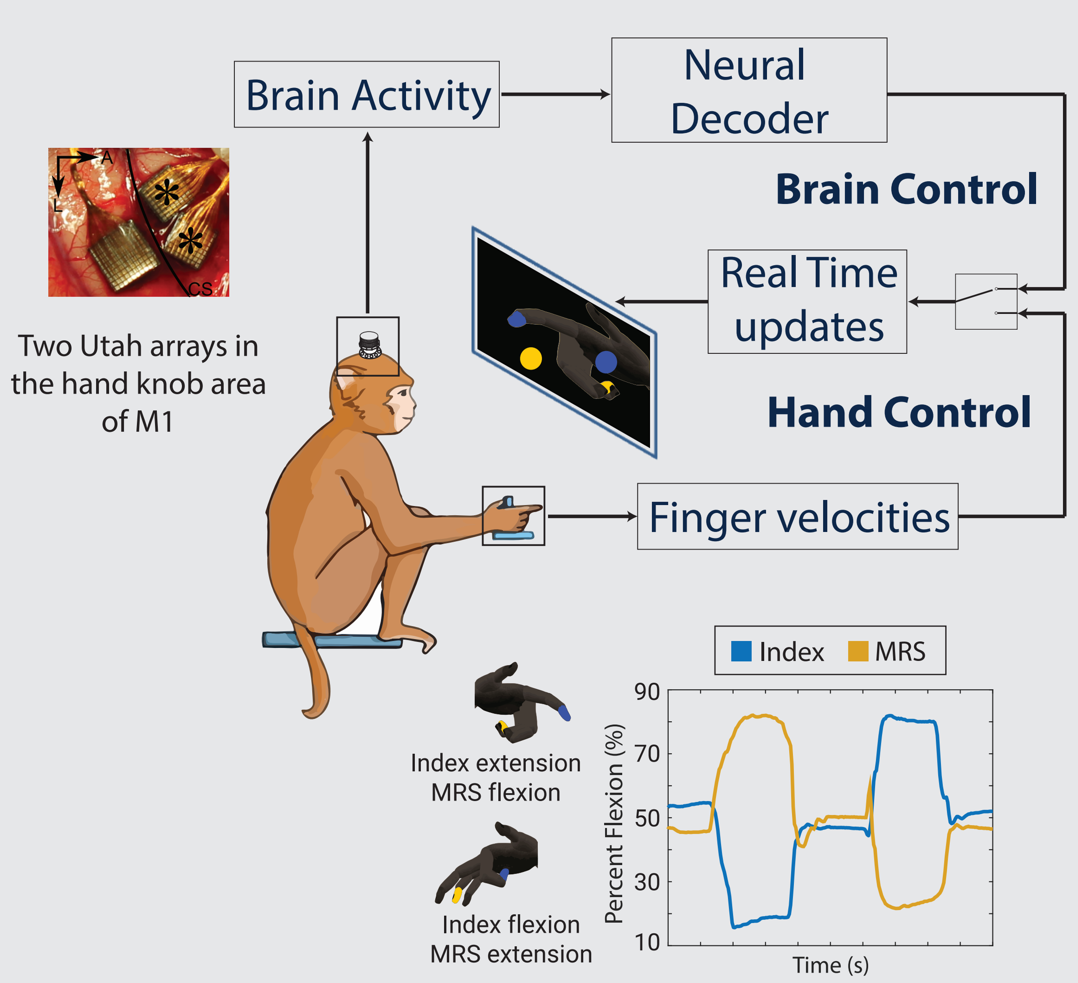
The **Kalman filter** is an attractive linear model, as it can use **knowledge about physics** of the movement to improve decoding and still maintain an **explainable structure**. However, as a linear model, it **cannot accurately model the complexity** of the brain signals.

Here, we tested **KalmanNet** [1], an algorithm that combines the **Kalman filter with deep learning**, to improve performance with **model mismatch** while keeping an **explainable structure.**

METHODS

We **implanted one monkey with a microelectrode array** (96 channels) in the hand area of the motor cortex and trained him to do a **two finger movement task** [2].

We **tested algorithms offline** (pre-recorded data) and **online** (real-time brain control) across multiple days.

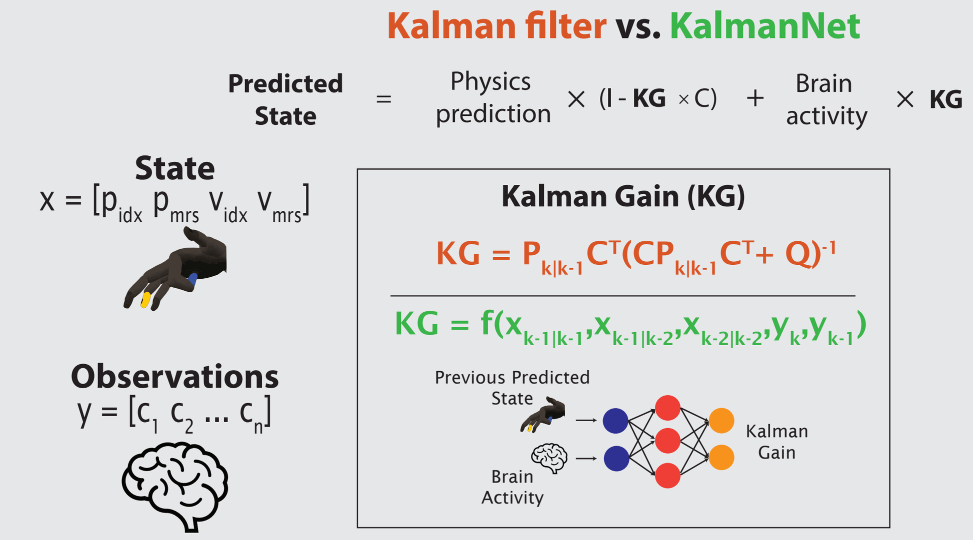


KALMANNET = KALMAN FILTER + DEEP LEARNING

KalmanNet differs from the Kalman filter in that it uses a **neural network to compute the Kalman gain**, which **guides how to merge the observations and the physics predictions**.

It uses **linear observation and trajectory models**, learned through a calibration run using least squares.

The network is composed of a series of **recurrent and feed-forward layers** and is trained indirectly with the state prediction error.

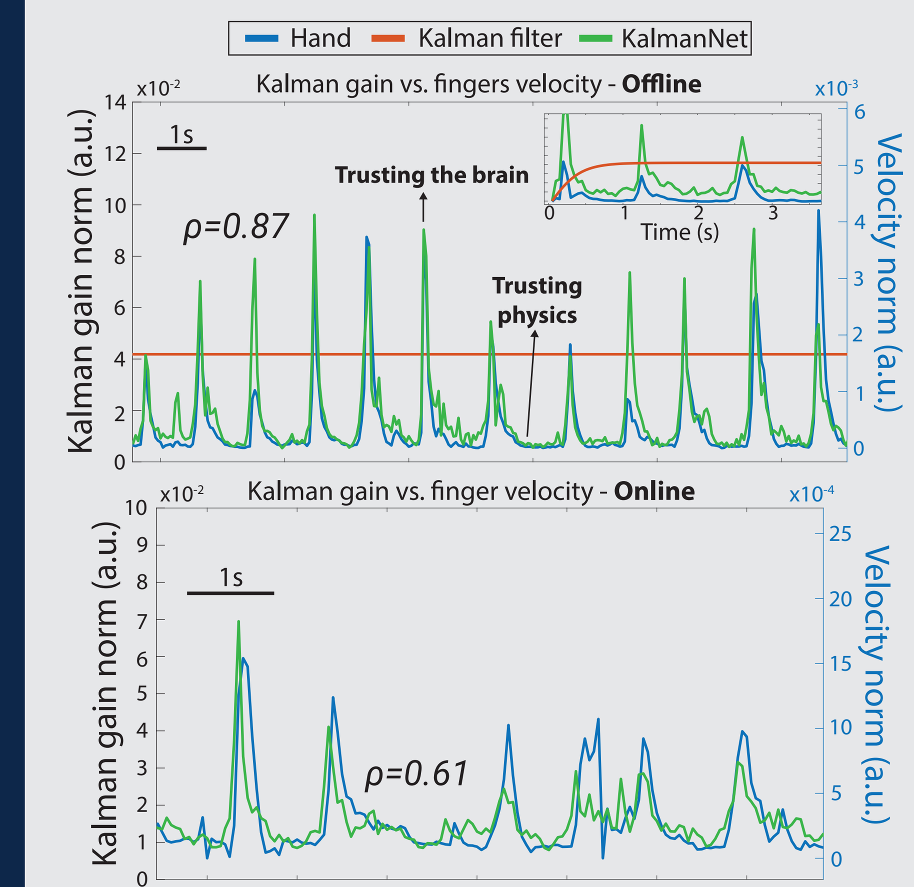


COMPARISON TO OTHER DECODERS

We compared the performance of four different decoding algorithms: **Kalman filter, KalmanNet, tcFNN** [3], **and LSTM** [4]**.**

The **tcFNN** model uses **time convolution and feed-forward layers** to predict velocity, while the **LSTM** predicts position and velocity by maintaining a **hidden state**.

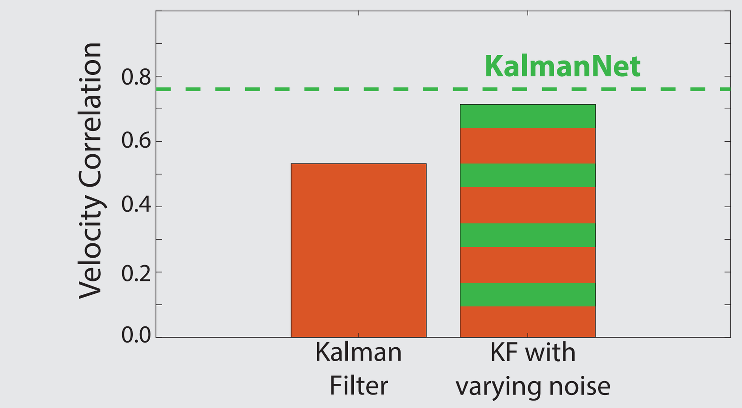
KALMANNET AS A NON-LINEAR TRUST SYSTEM



In KalmanNet, the Kalman gain norm co-varied with the finger velocity, indicating that the **model learned to trust the brain activity to generate high velocities.**

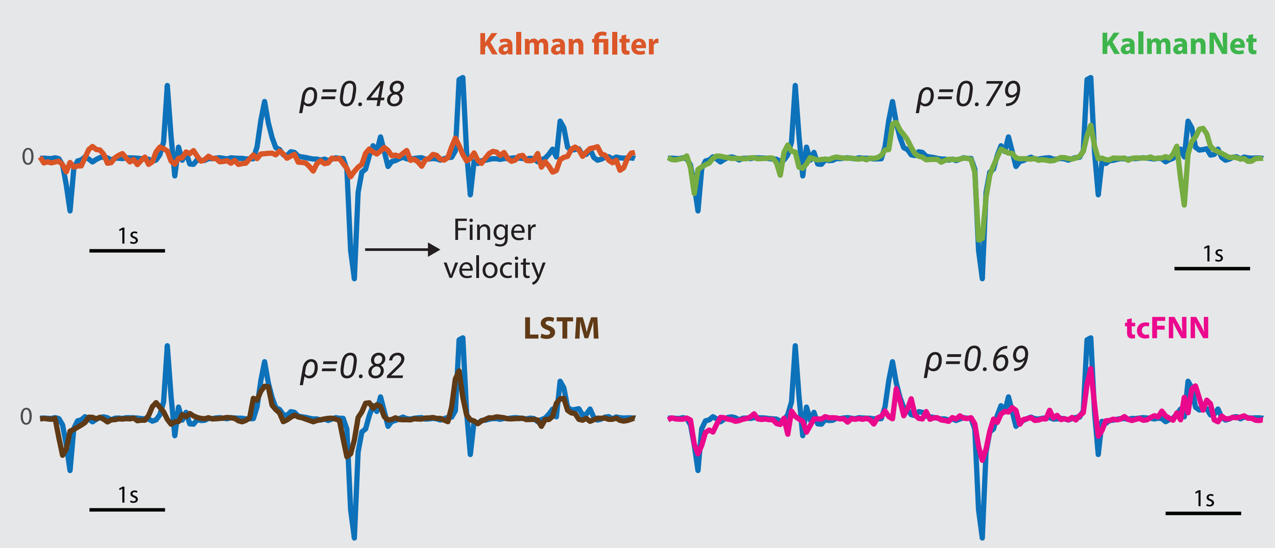
During online trials, this phenomenon was also present: the **Kalman gain covaried with the output velocity.**

KalmanNet's performance boost can be replicated in a KF by **"cheating" and making the noise model covary with the velocity.**

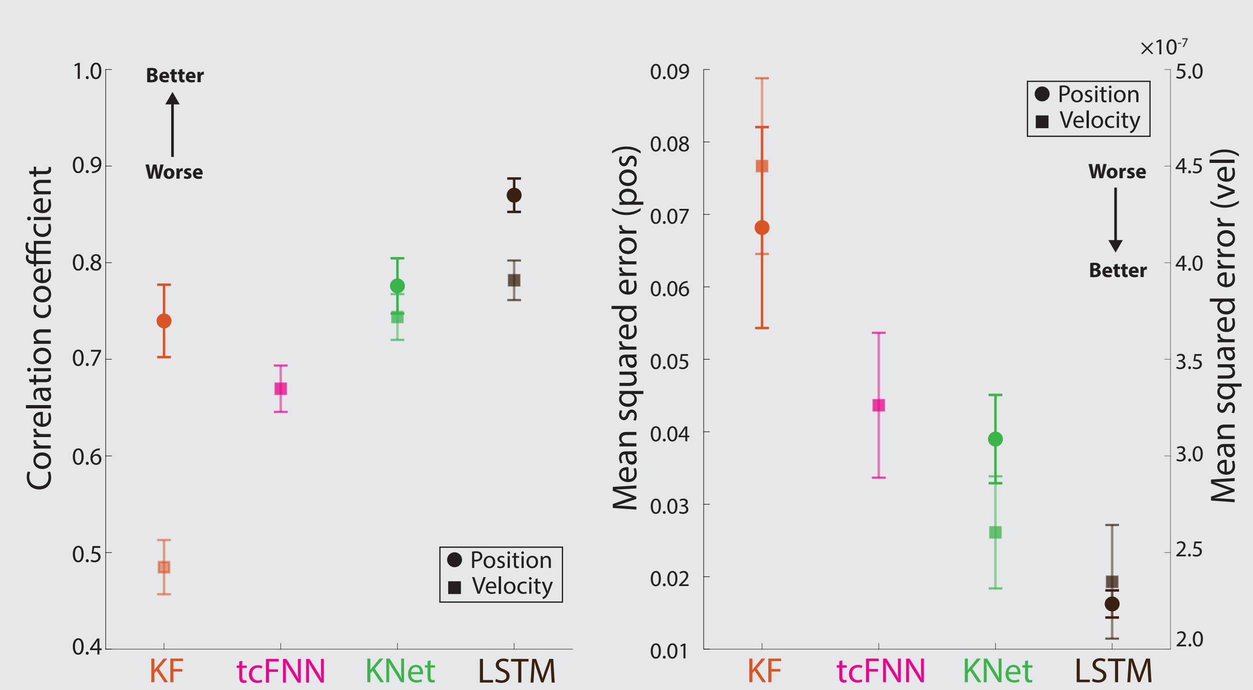


KALMANNET PERFORMS WELL OFFLINE

We tested all algorithms offline and used them to predict 2D finger position and velocity. The **Kalman filter performed poorly in low and high velocity regimes**, while KalmanNet, LSTM, and tcFNN matched pre-recorded kinematics better.



Across 9 days of offline testing, **KalmanNet was comparable to the LSTM in predicting position and velocity**. The order of performance between algorithms was the same when looking at correlation or MSE.



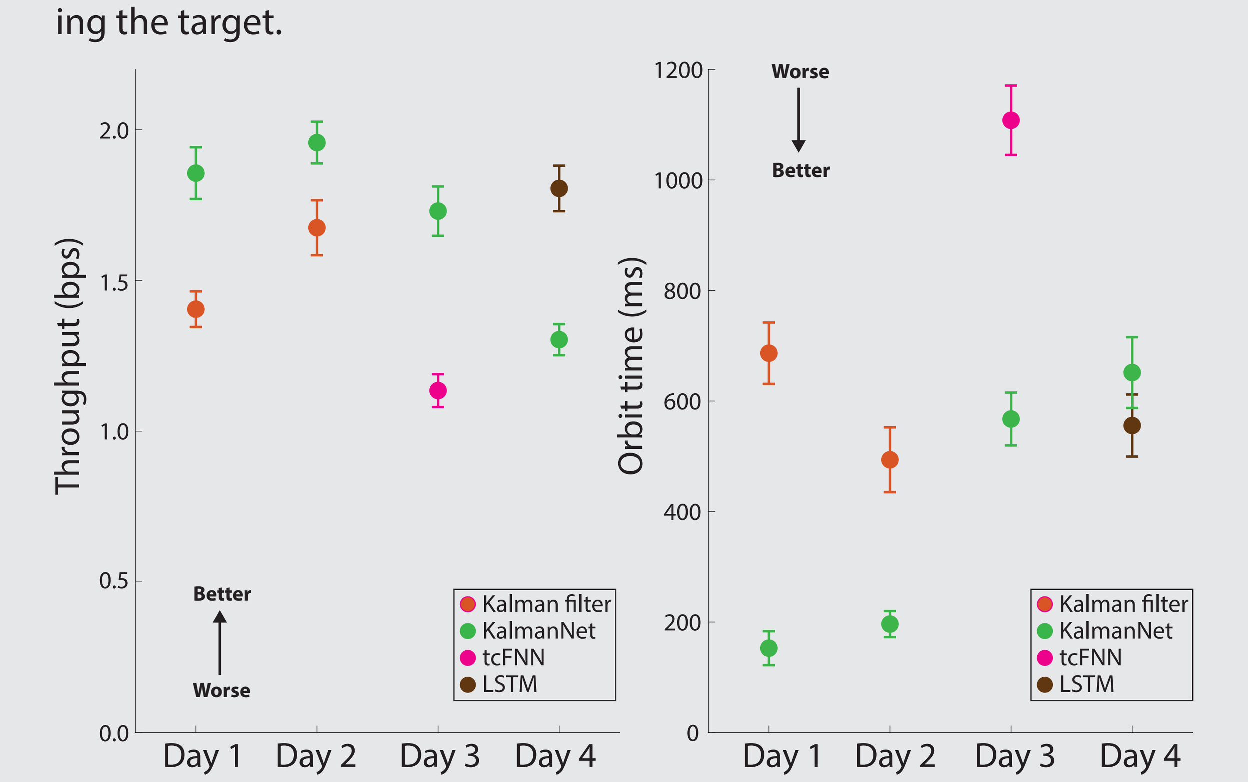
PRELIMINARY ONLINE PERFORMANCE

Snippet of **online trials of the Kalman filter and KalmanNet**. The Kalman filter is worse at stopping at the targets, which gets translated into **larger orbiting times** around the target.

Tablet Here

We tested the algorithms online on 4 different days in an **ABA manner**: 150 trials of algorithm A, 150 of algorithm B, and then 150 of algorithm A.

Throughput measures how fast a trial was completed, while taking into account its difficulty. **Orbiting time** corresponds to the time spent orbiting the target.



CONCLUSIONS

We demonstrated here that **KalmanNet**, an algorithm that combines the Kalman filter and deep learning, **can accurately predict movement from brain activity**, and that it **performed better than the Kalman filter and was comparable to other state-of-the-art decoders**, in offline and online settings.

KalmanNet worked by **modulating the trust of the system**, increasing trust in the brain measurements for high velocities, and trusting the physics model for low velocities.

Next steps include implementing **optimizations to improve performance**, and testing in different tasks, such as with **EMG decoding**.

REFERENCES AND ACKNOWLEDGEMENTS

[1] G. Revach et al., IEEE Transactions on Signal Processing 2022, 70, 1532.
[2] S. R. Nason et al., Neuron 2021, 109, 3164.
[3] M. S. Willsey et al., Nat Commun 2022, 13, 6899.
[4] J. T. Costello, H. Temmar, **L. H. Cubillos**, et al., bioRxiv, 2023.

We thank **Eric Kennedy**, ULAM, Veterinary Staff, and Chestek lab members for expert animal care and experimental assistance. This work was supported by NSF 1926576, Kahn Foundation 2029755, NSF Grant 1926576, NSF GRFP 1841052, NIH T32NS007222, UM Robotics Institute, and **ANID of Chile**.