KalmMind: A Configurable Kalman Filter Design Framework for Embedded Brain-Computer Interfaces

Guy Eichler
Dept. of Computer Science
Columbia University
New York, New York, USA
guyeichler@cs.columbia.edu

Joseph Zuckerman
Dept. of Computer Science
Columbia University
New York, New York, USA
jzuck@cs.columbia.edu

Luca P. Carloni
Dept. of Computer Science
Columbia University
New York, New York, USA
luca@cs.columbia.edu

***Abstract*—Kalman Filter (KF) is one of the most prominent algorithms to predict motion from measurements of brain activity. However, little effort has been made to optimize the KF for deployment in embedded brain-computer interfaces (BCIs). To address this challenge, we propose a new framework for designing KF hardware accelerators specialized for BCI, which facilitates design-space exploration by providing a tunable balance between latency and accuracy. Through FPGA-based experiments with brain data, we demonstrate improvements in both latency and accuracy compared to the state of the art.**

I. Introduction

Brain-computer interfaces (BCIs) are becoming increasingly complex, with the resolution and throughput of neural data growing exponentially as the number of electrodes in neural interfaces increases [1]–[3]. However, achieving an implant-based BCI system that supports mobility, real-time computation, and low power consumption remains a challenge.

The Kalman Filter (KF) is the most widely used algorithm for motion prediction in BCIs, proven effective in estimating movements of body parts from neural data [4]–[6]. Yet, existing KF implementations are not optimized for processing diverse, high-dimensional neural data or for meeting the strict power and latency constraints of embedded BCI systems [7], [8].

For this reason, we introduce KalmMind, a novel framework for designing configurable KF hardware accelerators tailored to BCI applications. KalmMind provides design flexibility, enabling control over computational intensity by allowing for a tunable balance between latency and accuracy. It leverages spatiotemporal correlations in neural activity and targets the primary bottleneck of the KF—the matrix inverse. We demonstrate how KalmMind facilitates the design of efficient and accurate KF hardware accelerators, unlocking new possibilities for real-time motion prediction in embedded BCI systems.

II. Background

Cutting-edge BCI systems integrate thousands of implanted micro-electrodes to record high-resolution neural data [3]. However, running BCI applications close to the brain presents significant challenges, as the sensitivity of brain tissue to heat limits the power consumption of implanted devices [7].

For this reason, BCI systems have been decomposed into two main components: an implanted chip and a wearable relay station [2]. The implanted chip records neural data and communicates wirelessly with the relay station. The relay station executes applications in real time and can consume only up to $\sim 200mW$ within the body-area network (BAN) [8].

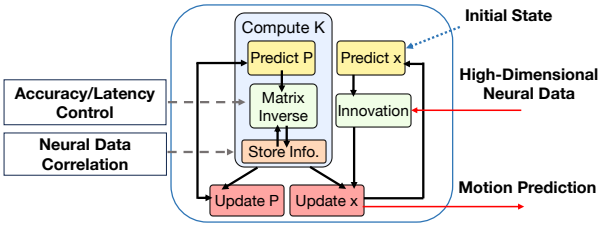

Fig. 1: The high-level architecture of a KalmMind accelerator.

TABLE I: The Accuracy of the KF with Different Methods

Accuracy Metric	Gauss	Taylor [11]	SSKF [12]	Newton [10]
MSE	3.8×10^{-12}	0.05	0.1	6.6×10^{-6}
MAE	7×10^{-7}	0.08	0.06	0.0004
*Max. Difference (%)	0.008	9.7×10^{2}	5.3×10^{2}	4
*Avg. Difference (%)	0.0001	9	4.8	0.035

*These scores are normalized with respect to the original KF output [4].

Kalman Filter Design for BCI. The Kalman Filter (KF) remains the primary motion decoding algorithm in BCI applications [4]–[6]. The KF is an iterative algorithm, with the computation of the Kalman gain (K) as a central step, and matrix inversion serving as its bottleneck. Most BCI applications that utilize the KF still rely on Gaussian elimination (Gauss), the standard method for calculating the matrix inverse. Although accurate, Gauss suffers from internal data dependencies that limit the performance of hardware implementations.

Our goal is to design optimized KF hardware accelerators that introduce less than $\sim$10% error compared to the standard KF using Gauss, thus ensuring precise control over fine motor tasks in BCI systems [9], while improving performance and enabling low power consumption.

TABLE II reports KF accuracies when integrated with various computational methods. For all methods, the KF predicts motion based on brain data from a non-human primate (NHP) over 100 iterations, with the output compared to that provided by Glaser *et al.* [4]. The methods other than Gauss offer different approximations of K. Among them, the Newton-Raphson method (Newton) [10], which we use to approximate the matrix inverse, provides the best accuracy.

III. KalmMind

Fig. 1 presents the main modules of the KalmMind KF and their dependencies. We isolate the computation of K, allowing for easy modification with different computation techniques.

978-3-9826741-0-0/DATE25/© 2025 EDAA

TABLE II: Accuracy Results with Three Neural Datasets

	MSE	MAE	MAX DIFF
Motor	2.1×10^{-13}–1.1×10^{-6}	2×10^{-7}–1.6×10^{-4}	4.3×10^{-5}–1.91
Soma.	2.2×10^{-13}–9.9×10^{-6}	2.3×10^{-7}–5.1×10^{-4}	3.5×10^{-5}–5.3
Hippo.	3.1×10^{-11}–7.1×10^{-11}	1.2×10^{-6}–2.2×10^{-6}	8.2×10^{-5}–2.1×10^{-3}
Baseline	4.8×10^{-13}, 3×10^{-13}, 3.5×10^{-11}	2.7×10^{-7}, 2.7×10^{-7}, 1.4×10^{-6}	1.1×10^{-4}, 8.5×10^{-5}, 3.8×10^{-4}

Computation of the Matrix Inverse. While generally accurate, direct *calculation* of the matrix inverse relies on floating-point divisions, which may introduce numerical errors. Moreover, data dependencies limit the ability to decompose the calculation and exploit parallel processing.

An *approximation* of the matrix inverse requires fewer computational steps and can reduce data dependencies, enabling more parallel processing [10]. Iterative approximations commonly do not involve divisions, making them less prone to numerical errors. However, selecting an *initial seed* close to the optimal matrix inverse is crucial for convergence.

Combining Calculation and Approximation. Neural datasets are highly diverse, as they are recorded from different brain regions and neural interfaces, each requiring varying levels of computational accuracy and real-time performance.

We propose a new, simple yet powerful technique that interleaves two methods between consecutive KF iterations: one for calculation and the other for approximation. This technique balances a highly accurate but slow calculation with a faster, less accurate approximation. At each KF iteration, one of the two methods is selected. For approximation, we use the inverse matrix computed from previous measurements as the initial seed for the current iteration, leveraging the strong temporal and spatial correlations between neural data measurements [1].

IV. Experimental Evaluation

Methodology: KalmMind accelerators expose memory-mapped registers for controlling communication with main memory and matrix operations. These registers configure the dimensions of matrices and vectors, set the number of KF iterations, select the data path for matrix inversion, and set the number of internal iterations during approximation. We synthesize hardware accelerators, supporting a 32-bit floating-point data type, by using Vivado, and leverage ESP [13] to design FPGA-based SoCs capable of running custom Linux software applications that invoke the accelerators. We conduct experiments on the XCVU440 FPGA @78MHz and test three neural datasets from distinct brain regions [4].

Results: We demonstrate KalmMind with an accelerator that implements Gauss calculation and Newton approximation (Gauss/Newton), consuming $185mW$. We compare against a Gauss-only implementation (baseline) and compute three accuracy metrics: *(i)* Mean Squared Error (MSE), *(ii)* Mean Absolute Error (MAE), and *(iii)* the maximum difference between an output value and its expected value (MAX DIFF).

TABLE II summarizes the accuracy ranges to which the accelerator can be configured for each dataset and metric after 100 KF iterations. Thanks to the flexibility of the accelerator, we can adjust the computation across a broad range of accuracies. In all cases, we find configurations that even outperform the baseline in terms of accuracy. With the same set of configurations, each dataset achieves different accuracy ranges.

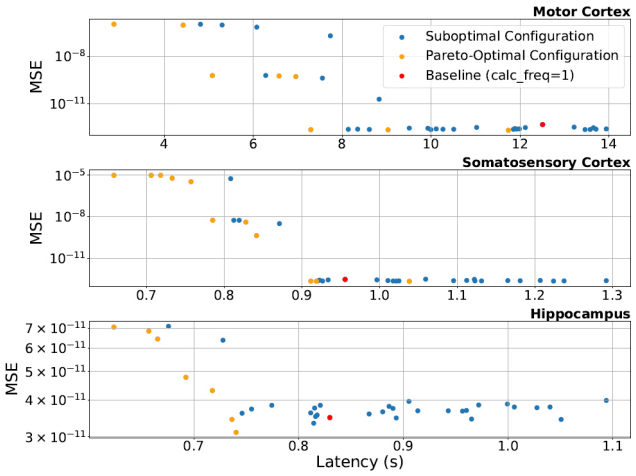

Fig. 2: Latency vs. accuracy with the Gauss/Newton accelerator.

We combine the accuracy analysis with latency measurements when running on FPGA. Fig. 2 shows the MSE for each dataset, with each Pareto-optimal point representing a specific trade-off between accuracy and latency. We identify configurations that achieve up to 55% better accuracy and up to $4.4\times$ speedup compared to the baseline, resulting in improved KF performance. These gains make KalmMind an ideal solution for real-time BCI applications.

V. Conclusion

KalmMind is the first framework for designing configurable KF hardware accelerators, offering fine-grained control over latency and accuracy to address the diversity of brain data and to accommodate the constraints of real-world BCI systems. In future work, KalmMind will integrate more calculation and approximation methods, thereby expanding the design space and providing greater flexibility for various BCI applications.

References

[1] A. E. Urai *et al.*, "Large-Scale Neural Recordings Call for New Insights to Link Brain and Behavior," *Nature neuroscience*, 2022.

[2] N. Zeng *et al.*, "A Wireless, Mechanically Flexible, 25m-Thick, 65,536-Channel Subdural Surface Recording and Stimulating Microelectrode Array with Integrated Antennas," in *VLSI*, 2023.

[3] Y. Wang *et al.*, "Implantable Intracortical Microelectrodes: Reviewing the Present with a Focus on the Future," *MNE*, 2023.

[4] J. I. Glaser *et al.*, "Machine Learning for Neural Decoding," *Eneuro*, 2020.

[5] A. D. Degenhart *et al.*, "Stabilization of a Brain–Computer Interface via the Alignment of Low-dimensional Spaces of Neural Activity," *Nature biomedical engineering*, 2020.

[6] X. Zhang *et al.*, "Reinforcement learning-based Kalman filter for adaptive brain control in brain-machine interface," in *EMBC*, 2021.

[7] P. D. Wolf *et al.*, "Thermal Considerations for the Design of an Implanted Cortical Brain–Machine Interface (BMI)," *Indwelling Neural Implants: Strategies for Contending with the In Vivo Environment*, 2008.

[8] G. Eichler *et al.*, "MasterMind: Many-Accelerator SoC Architecture for Real-Time Brain-Computer Interfaces," in *ICCD*, 2021.

[9] Z. Irwin *et al.*, "Neural Control of Finger Movement via Intracortical Brain–machine Interface," *J. Neural Eng.*, 2017.

[10] V. Pan *et al.*, "Efficient parallel solution of linear systems," in *STOC*, 1985.

[11] Y. Liu *et al.*, "Efficient Mapping of a Kalman Filter into an FPGA Using Taylor Expansion," in *FPL*, 2007.

[12] W. Q. Malik *et al.*, "Efficient Decoding with Steady-State Kalman Filter in Neural Interface Systems," *IEEE TNSRE*, 2010.

[13] P. Mantovani *et al.*, "Agile SoC Development with Open ESP," in *ICCAD*, 2020.