

Eric Christiansen, PhD

☎ +1 (202) 957-3618 | ✉ eric@emchristiansen.com | 🐙 GitHub | 🎓 Scholar | 🔗 LinkedIn | 🌐 Website

*Software engineer experienced in AI research and infrastructure.
See emchristiansen.com for details.*

WORK AND RESEARCH

High-frequency crypto arbitrage

Solopreneur

2022 July – Present

Created a market-neutral high-frequency cryptocurrency arbitrage business with \$300K profit on \$10M volume. Notable subtasks:

- Wrote SOTA CUDA code for fast traversal of the Curve25519 elliptic curve.
- Created smart contracts to house logic that must be run during block creation.
- Co-designed a fast dynamic-programming algorithm to induce desired transaction orderings in new blocks.
- Generalized trading strategies into a differentiable flow model, implemented in JAX and bound to Rust via PyO3.

Used Typescript, OCaml, Rust, Python, CUDA, Kubernetes, JAX, and TiDB. Code is available for in-meeting review upon request.

Google Research

Senior Software Engineer

Mountain View, CA

2014 April – 2022 July

Work split between AI research and internal infrastructure, under [Philip Nelson](#), [Kevin Murphy](#), and [Sergey Ioffe](#). Selected projects:

- Led an 18-person team which developed a seminal AI computational microscopy technique, published in *Cell* [1], the [Google Blog](#), and [open sourced](#). Work ranged from wetlab experimental design, to large scale distributed image processing, to the design of a novel image-to-image model. It was patented [2] and led to the creation of two new projects at Verily. Later work used deep learning to automate quality control in similar pipelines [3]. See [this editorial](#) for context.
- Created Google's first hyperparameter tuning API for deep learning, which was later rewritten and made a cloud product.
- Created a system to speed training of deep networks by dynamically and automatically adjusting the train set data distribution, leading to 30% faster training on tasks with imbalances in example difficulty.
- Published on neural architecture search [4, 5] and model calibration and ensembling [6, 7].
- Maintained and developed TensorFlow and JAX libraries for calibration and ensembling, nondifferentiable optimization, and tensor manipulation.
- Hosted 3 successful interns, interviewed 100s of candidates, earned designations for cross-team collaboration and technical expertise, got readability for Golang, C++, and Python, and finished my PhD [8]

Note, I quit six months before the first layoffs in 2023.

Willow Garage

PhD Intern

Menlo Park, CA

2012 – 2013 (9 months)

Worked in robot perception.

- Developed two local image descriptors designed for speed on mobile devices [9] (C++ and Scala).
- Created an open-source evaluation framework for local descriptors (Scala).
- Added Java support to OpenCV.
- Taught fitness classes and organized company-wide meals.

Google

PhD Intern

Mountain View and Los Angeles, CA

2011 and 2010 Summers (6 months)

- Worked on Google Goggles research and backend infrastructure, including adding the first high-dimension log time nearest neighbor method to the scalable matching service.
- Helped the webcrawler to detect and appropriately handle auto-generated websites.

EDUCATION

University of California, San Diego

PhD in CS

La Jolla, CA

2008 – 2018

PhD in computer science, with foci in computer vision and machine learning.

- Published in ML theory [10], computer vision [11, 12, 13, 14, 10], and deep learning [1].
- TA'd undergrad computability and graduate algorithms (2x), graduate data mining, and graduate computer vision. Mentored Google Summer of Code student for OpenCV.
- Started a free food program for my lab, which I believe significantly improved lab morale and communication.

Swarthmore College

BA in Math (honors) with CS minor

Swarthmore, PA

2004 – 2008

CODE PROJECTS

During my PhD, I kept sane by working on a number of side-projects, for example:

- [PersistentMap](#): A type-safe, boilerplate-free, key-value store for Scala.
- [salve](#): A macro and template library for adding some functional programming ideas to C++.
- [sbt-latex](#): A build management tool for LaTeX.
- [CharikarLSH](#): An implementation of Moses Charikar's method for approximate nearest neighbor retrieval, in C++. Note, techniques like this are how vector databases work.
- [mbtree](#): An implementation of metric-ball trees for nearest neighbor search, in Scala.
- [DistanceLSH](#): An implementation of a metric hashing for nearest neighbor search, in Haskell.

MISCELLANEOUS

Things I do sometimes

- CrossFit
- Running
- Burning Man
- [AIDS/LifeCycle](#)

Books I liked

- [Harry Potter and the Methods of Rationality](#)
- [The Traitor Baru Cormorant](#)
- [The Selfish Gene](#)
- [Sapiens](#)

Non-research interests

- Ethereum
- [Tezos](#)
- Rust

REFERENCES

- [1] E. Christiansen, S. J. Yang, D. M. Ando, A. Javaherian, G. Skibinski, S. Lipnick, E. Mount, A. O'Neil, K. Shah, A. K. Lee *et al.*, "In silico labeling: Predicting fluorescent labels in unlabeled images," *Cell*, 2018.
- [2] P. C. Nelson, E. M. Christiansen, M. Berndl, and M. Frumkin, "Processing cell images using neural networks," May 15 2018, uS Patent 9,971,966.
- [3] S. J. Yang, M. Berndl, D. Michael Ando, M. Barch, A. Narayanaswamy, E. Christiansen, S. Hoyer, C. Roat, J. Hung, C. T. Rueden *et al.*, "Assessing microscope image focus quality with deep learning," *BMC bioinformatics*, vol. 19, pp. 1–9, 2018.
- [4] A. Klein, E. Christiansen, K. Murphy, and F. Hutter, "Towards reproducible neural architecture and hyperparameter search," 2018.
- [5] C. Ying, A. Klein, E. Christiansen, E. Real, K. Murphy, and F. Hutter, "Nas-bench-101: Towards reproducible neural architecture search," in *International conference on machine learning*. PMLR, 2019, pp. 7105–7114.
- [6] X. Wang, D. Kondratyuk, E. Christiansen, K. M. Kitani, Y. Movshovitz-Attias, and E. Eban, "On the surprising efficiency of committee-based models," *arXiv preprint arXiv:2012.01988*, 2020.

- [7] X. Wang, D. Kondratyuk, E. Christiansen, K. M. Kitani, Y. Alon, and E. Eban, “Wisdom of committees: An overlooked approach to faster and more accurate models,” *arXiv preprint arXiv:2012.01988*, 2020.
- [8] E. Christiansen, “From local descriptors to in silico labeling,” Ph.D. dissertation, University of California, San Diego, 2018.
- [9] E. Christiansen, V. Rabaud, A. Ziegler, I. Essa, D. Kriegman, and S. Belongie, “Match-time covariance for descriptors,” in *British Machine Vision Conference (BMVC)*, 2013.
- [10] E. Christiansen, “An upper bound on prototype set size for condensed nearest neighbor,” *arXiv preprint arXiv:1309.7676*, 2013.
- [11] T. Winlock, E. Christiansen, and S. Belongie, “Toward real-time grocery detection for the visually impaired,” in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*. IEEE, 2010, pp. 49–56.
- [12] A. Ziegler, E. Christiansen, D. Kriegman, and S. Belongie, “Locally uniform comparison image descriptor,” *Advances in Neural Information Processing Systems*, vol. 25, 2012.
- [13] E. Christiansen, I. S. Kwak, S. Belongie, and D. Kriegman, “Face box shape and verification,” in *International Symposium on Visual Computing (ISVC)*, 2013.
- [14] A. Flores, E. Christiansen, D. Kriegman, and S. Belongie, “Camera distance from face images,” in *International Symposium on Visual Computing (ISVC)*, 2013.