

## Siyuan Chai

---

CONTACT INFORMATION	<a href="mailto:siyuanc3@illinois.edu">siyuanc3@illinois.edu</a> <a href="https://schai.me">https://schai.me</a>
EDUCATION	<b>University of Illinois</b> , Urbana Champaign, IL Computer Science Ph.D. Start Aug. 2021 Advisor: <a href="#">Prof. Tianyin Xu</a> <b>Northwestern University</b> , Evanston, IL M.S. Computer Science, B.S. Electrical Engineering Graduated June 2021 GPA: 4.0/4.0 (Summa Cum Laude)
PUBLICATIONS	<ol style="list-style-type: none"><li>1. Brian Suchy, Souradip Ghosh, Aaron Nelson, Zhen Huang, Drew Kersnar, <b>Siyuan Chai</b>, Michael Cuevas, Gaurav Chaudhary, Alex Bernat, Nikos Hardavellas, Simone Campanoni, Peter Dinda. "CARAT CAKE: Replacing Paging via Compiler/Kernel Cooperation." <i>To appear in ASPLOS 2022</i>.</li><li>2. Ramsey M Wehbe, Jiayue Sheng, Shinjan Dutta, <b>Siyuan Chai</b>, Amil Dravid, Semih Barutcu, Yunan Wu, Donald R. Cantrell, Nicholas Xiao, Hatice Savas, Rishi Agrawal, Nishant Parekh, Aggelos K. Katsaggelos. "Deepcovid-xr: An artificial intelligence algorithm to detect covid-19 on chest radiographs trained and tested on a large us clinical dataset." <i>Radiological Society of North America</i>. [Online]. Available: <a href="https://doi.org/10.1148/radiol.2020203511">https://doi.org/10.1148/radiol.2020203511</a>.</li></ol>
RESEARCH EXPERIENCE	<b>UIUC Xlab</b> , <a href="#">Prof. Tianyin Xu</a> Aug. 2021 to Present <i>Support Linux Kernel for Elastic Cuckoo Page Table</i> <ul style="list-style-type: none"><li>• Adapting Linux kernel, primarily the memory management portion, to support <a href="#">Elastic Cuckoo Page Table</a> (ECPT), a hash page table that aims to replace paging by enabling memory-level parallelism</li><li>• Extensively modified memory translation portion of QEMU to simulate ECPT's hardware behavior</li><li>• Exploring the design space like page management, allocation and cache policy in linux running on ECPT</li></ul> <b>NU Compilers Group</b> , <a href="#">Prof. Simone Campanoni</a> Jan. 2021 to July 2021 <i>Enhance Parallelism by Utilizing Commutative Loop iterations</i> <ul style="list-style-type: none"><li>• Coded a LLVM pass to tell the commutativity of &lt;load, ALU operation, store&gt; triplet across loop iterations</li><li>• Extend the idea to develop tools for loop iteration commutativity for further utilization of parallelism</li></ul> <b>NU Parallelism Group</b> , <a href="#">Prof. Peter Dinda</a> June 2020 to May 2021 <i>CARAT CAKE: Replacing Paging via Compiler/Kernel Cooperation</i> <ul style="list-style-type: none"><li>• Designed and implemented CARAT CAKE, an allocation level address space which aims to replace virtual memory and paging with protection checks inserted at compile time and allocations tracked in runtime</li><li>• Implemented a competitive paging address space with support for red black tree and splay tree data structures to track VA-PA mapping, Transparent Huge Pages, and PCID; performance measured with Performance Monitoring Counter</li><li>• Designed runtime protection check with address mapping data structures</li></ul> <b>Image &amp; Video Processing Lab</b> , <a href="#">Prof. Aggelos Katsaggelos</a> June 2019 to July 2021 <i>DeepCOVID-XR</i> <ul style="list-style-type: none"><li>• Designed and implemented a CNN model to flag out positive COVID cases based on patients' chest X-ray images</li><li>• Outperformed experienced radiologists with an accuracy of 85% compared to 76 - 82% and AUC of 0.935 compared to 0.819 - 0.856</li></ul>

WORK EXPERIENCE	<b>Research Intern</b> , Tencent Network Group <i>Service Driven Network Verification tool</i> <ul style="list-style-type: none"> <li>Contributed to design a scalable network verification that supports quantitative query and covers all data plane with global formal modeling and local simulation</li> <li>Designed easy-to-use geo-based intent language for network verification</li> </ul>	June 2021 to Aug. 2021
PROJECTS	<b>C-style Language Compiler</b> , CS 322 Compiler Construction <ul style="list-style-type: none"> <li>Created, from scratch, a compiler to translate C-style language to x86_64 assembly</li> <li>Implemented features including graph-coloring register allocation, liveness analysis, instruction selection with tiling, control flow graph, and memory access checking</li> </ul> <b>Middle End Analysis for a C-based API</b> , CS 323 Code Analysis & Transformation <ul style="list-style-type: none"> <li>Coded a LLVM pass to reduce calls to a custom C-based API by implementing analysis like reaching-definition, constant propagation and folding, alias analysis for the specific API, function inlining, and dead code elimination</li> </ul>	
SKILLS	<b>Programming languages:</b> C/C++, Assembly, Python, Java, Go, JavaScript, MATLAB <b>System-level Development:</b> Unix/Linux, QEMU, Docker, GDB, Make, Linker, LLVM, OpenMP <b>Artificial Intelligence:</b> CUDA, PyTorch, Tensorflow, Keras, Image Processing, Computer Vision <b>Hardware:</b> Raspberry Pi, Arduino, VHDL, Verilog <b>Web Development:</b> HTML, CSS, Flask, Django, React	
PROFESSIONAL ACTIVITIES	<b>SOSP 2021:</b> Artifact Evaluation Committee, Slack Co-chair	
TEACHING EXPERIENCE	<b>Peer Mentor (Undergraduate TA)</b> - Northwestern University Spring 2021 CS 336 - Design & Analysis of Algorithms with <a href="#">Prof. Jason Hartline</a> Winter 2021 CS 343 - Operating Systems with <a href="#">Prof. Peter Dinda</a> Winter 2020 CS 336 - Design & Analysis of Algorithms with <a href="#">Prof. Konstantin Makarychev</a> Fall 2019 CS 336 - Design & Analysis of Algorithms with <a href="#">Prof. Jason Hartline</a> Spring 2019 CS 336 - Design & Analysis of Algorithms with <a href="#">Prof. Jason Hartline</a> <b>Teaching Assistant</b> - Washington University in St. Louis Spring 2018 ESE 205 Introduction to Engineering Design with <a href="#">Prof. James Feher</a>	
AWARDS AND HONORS	<b>Dean's List</b> , all quarters ACM-ICPC, Mid-Central Regional, <b>Top 20%</b> VEX Robotics International Championship, <b>Top 4 Alliance</b>	2017 - 2021 2018 2016