Ze Ming (Tommy) Xiang

 github.com/TommyX12 | Toronto ·ON
tommy.xiang@mail.utoronto.ca | (647)-968-4242

Skills

Languages: **Python**, **Java**, **JavaScript**, **C/C++**, **TypeScript**, HTML, CSS, C#, QML, SQL, Haxe, Lua, Verilog, Bash, Emacs Lisp.
Frameworks: **PyTorch**, **Angular 2**, Node.js Express, Ionic, React, Bazel, Vue.js, TensorFlow, OpenFL, Qt Quick.
Tools: Git, Perforce, Emacs, Vim, Linux, Unity3D, G-Suite, Microsoft Office.

Education

University of Toronto | Computer Science Specialist | Honors Bachelor of Science   **Class of 2021**
CGPA: 3.98/4.0, Dean's List Scholar every semester, President's Entrance Scholarship

Upper-level Courses:	Neural Networks and Machine Learning	Algorithm Design & Complexity	Software Engineering
	Natural Language Computing	Introduction to Visual Computing	Computer Graphics

Professional Experience

- **NVIDIA**, *Deep Learning Research Intern*, Toronto Artificial Intelligence Lab, supervised by **Sanja Fidler**   2019 Fall - Present
  - Currently leading on-going research project for 3D asset generation from 2D images with minimal 3D supervision, using techniques including convolutional neural network, differentiable renderer, generative adversarial network, and Learned Perceptual Image Patch Similarity (LPIPS) loss
  - Achieved good qualitative results, including accurate inference of mesh shape and texture trained from *nuScenes* dataset
  - Developed modularized implementation of state-of-the-art point cloud classifier and differentiable renderer for **NVIDIA Kaolin**, the PyTorch library for 3D Deep Learning research, which gained more than **1,000 stars** on GitHub within 10 days after launch, with co-authored **paper submitted** to *arXiv*
  - Technologies: **PyTorch, Python, CUDA, OpenCV, NVIDIA GPU Cloud, Docker, CNN, GAN**

- **Google LLC**, *Software Engineering Intern*, Growth and Analytics   2019 Summer
  - Designed and implemented **full-stack** data quality alert system for internal intelligence visualization tool used by more than **30,000** Google employees
  - Deployed into production and immediately used to notify users of critical data anomalies, saving significant cost from **25-35** data errors annually
  - Designed and engineered anomaly detection system capable of automatically extracting and analyzing data points directly from frontend code
  - Technologies: **TypeScript, HTML/CSS, Angular 2, Java, Node.js, Python**

- **Google LLC**, *Software Engineering Intern*, Google Docs Team   2018 Summer
  - Implemented **MVC structured**, accessibility-friendly user interface for layout formatting on Google Docs Android, reaching over **100 million users**
  - Discovered and proposed solution for flaws in code base that will potentially cause 30% increase in typing latency
  - Technologies: **JavaScript, Java + Android, Google Closure, Bazel**

Technical Projects

- **Cellular** *2D Procedurally-Generated Game*   **C/C++, OpenGL, Lua, SDL 2, Cpp-Taskflow, Fruit, Entt**
  - Designed and developed procedurally-generated 2D action & adventure game, achieved **1st place** in UofT Game-Making Deathmatch 2017
  - Implemented a high-performance, ECS-based, multi-threaded game engine in **C/C++** and **OpenGL**. Designed concurrent renderer with fully automatic sprite batching. Designed lock-free task graph system allowing near-maximum hardware utilization using topological sorting and RLF vertex coloring heuristic. Utilized template metaprogramming to auto-detect race-condition. Achieved simulation of up to **200000+** animated entities at **60fps**
  - Designed and implemented automatic dependency injection framework in embedded Lua. Implemented procedural map generation using diamond-square algorithm, Whittaker diagram, cellular automata, and random walk; Developed culling algorithm to support seamless **100000+ tiles** map

- **Catalyzer** *Hybrid App for Time Management*   **TypeScript, Angular 2, Ionic, HTML, CSS**
  - Implemented time-management hybrid app using **TypeScript**, leveraging **Angular 2** and Ionic, with automatic scheduling and reward system
  - Developed real-time constraint-based optimal schedule generation through ad-hoc greedy algorithms, tree-based ordered sets, and priority queues
  - Achieved generation of **365** days of future schedule with respect to **3000** constraints within **0.5s** on a mobile device

- **Emacs Client for TabNine** (github.com/TommyX12/company-tabnine)   **Emacs Lisp**
  - Developed Emacs client for Jacob Jackson's code-completion system *TabNine*, which displays intelligent and relevant completions using deep learning
  - Implemented low-latency sub-process communication for candidate fetching, achieving **5ms** overhead per keystroke
  - Received more than **331 stars** on GitHub, and more than **2198** downloads on Emacs package archive

- **ShareSchedule** *Course Time-Table Generator*   **JavaScript, Node.js + Express, PostgreSQL**
  - Developed web application and RESTful API for UofT time-table generation using Node.js + Express and PostgreSQL
  - Implemented backtracking constraint-satisfaction algorithm on pure client-side JS to solve for conflict-free schedules with **<100ms** latency
  - Interface with Cobalt-UofT API to retrieve relevant course info, as well as using Facebook login to allow time-table sharing between students

Awards and Contributions

- 1st Place - Bloomberg Codecon UofT   2017
- 2nd Place - Microsoft Code Competition UofT   2017
  - Solved one of the hardest problem
- 2nd Best Accuracy - (National) USC Competition   2017
  - Developed geo-tagging tool for drone mission
- Silver Medalist -
  (National) Canadian Computing Olympiad (Link)   2016
- Co-President of
  Game Design and Development Club   2017 - 2018

- 1st - UofT Game-Making Deathmatch   2017
  - Best Overall and Best Technical Achievement Award
- 3rd Place - Big Data Challenge   2016
  - Analyzed and visualized open data using Python
  - Journal Published on STEM Fellowship (Link)
- Vision Subdivision Lead of
  University of Toronto Aerospace Team:
  Aerial Robotics division   2017 - 2018