YONG CHENG

Saint Louis, MO 63105 | yon.cheng@wustl.edu | +1 314-546-3998 | https://cocoyard.github.io/yong/

EDUCATION

Washington University in St. Louis — St. Louis, MO
- *Master of Science in Computer Science* — *August 2021 ~ May 2023*
- *GPA: 3.9/4.0*

Shanghai Ocean University — Shanghai, China
- *Bachelor of Science in Information and Computing Science* — *September 2016 ~ Jun 2021*
- *GPA: 3.9/4.0*

SKILLS

***Technical Skills*:** C/C++, Java, Python, Html/CSS, JavaScript (React, Node.js/NPM, Socket.IO, Vue, jQuery, Ajax), JSON, PHP, C#, MySQL/PostgreSQL/Firebase, Mathematica, Swift, Matlab, LaTeX
***Software/Technology*:** Git, AWS, Azure Blob Storage, Photoshop, Unity, Agile Software Development, RESTful API

WORK EXPERIENCES

Washington University in St. Louis | St. Louis, MO — *08/2022 ~ 12/2022*

Teaching Assistant for CSE 332/504N Object-Oriented Software Development Laboratory | **C++**

- Graded students' assignments based on quality, which were all labs in C++, where I did code reviews and pushed feedbacks to GitHub.
- Held office hours. Helped students with studio/lab questions by elaborating the reason behind the answer or asking them a similar but easier question to give them an inspiration to the original question.

Elekta | St. Louis, MO — *05/2022 ~ 08/2022*

Software Engineering Intern | **JavaScript, Node.js, Azure blob storage, PostgreSQL, Docker**

- Worked on an agent software for radiation therapy, which is used to manage DICOM files, a medical image standard such as CT photos.
- Implemented JavaScript to grant the software the ability to visualize DICOM files, to parse DICOM files, to recursively index attributes, to query DICOM files, and to upload and to download files.
- Configured and managed a database and a blob storage for users to put and fetch data. Utilized the framework developed by the company to build UI and optimized the interaction experience of that. Implemented RESTful API according to the design documentation.

PROJECTS (all in GitHub at https://github.com/CocoYard)

Microtubule tracking tool | Python, Napari, Image/video Processing — *12/2022*

- Microtubules are major components of the cytoskeleton. I was in charge of developing a tool to track their segmentations for biologists at *WashU*. Given the first frame's position of the selected microtubule, the tool should output its segmentation at all frames.
- Blurred the image and used *Contrast Limited Adaptive Histogram Equalization (CLAHE)* to adjust the global contrast. Designed algorithms to produce a binary image with dynamical threshold which is more accurate than the built-in function in *Open-CV*.
- Used algorithms such as *opening, closing, Hough transformation* and designed a loss function to get the target microtubule. Solved detection difficulties such as stretching, shrinking, moving, rotating, and crossing of the microtubules.

Sync Music Player | (full stack) HTML/CSS, JavaScript, React, Firebase, Node.js — *04/2022*

- An online music player which helps two people listen to music synchronously.
- Used Firebase to store user information and music state. Emit an event and update the corresponding front-end components whenever any user changes the pause/resume status or switches songs, so that the other user can see the same effect on his/her own page.

Calendar | (full stack) HTML/CSS, JavaScript, PHP, MySQL, jQuery — *03/2022*

- The whole page uses JavaScript's asynchronous request so that it partially downloads data from server instead of entirely redirecting to other pages. Users can add events to a specific date and create groups to share events together. The website passes the security checking (XSS, SQL injection, session hijack) and format checking (W3C HTML and CSS validators). It earned the top 5% grade among the class.

Pedagogical Applet of Chan's Algorithm | HTML/CSS, JavaScript, Vue — *12/2021*

- Developed a demo to display the implementation procedure of a convex hull algorithm, Chan's algorithm, which is efficient for the time complexity of O (n log *h*) where 'n' is the number of input vertices and *'h'* is the number of output vertices. This demo is used for the course CSE 546T Computational Geometry teaching in our school.
- Used *JavaScript* to conduct the algorithm and *Vue* to construct the sketch board. The demo stops at every necessary step for displaying the hull's construction of that stack frame. Designed an "auto" button where if it is pressed, the demo goes through all the steps spontaneously.

RESEARCH & CONTEST EXPERIENCES

Feature extraction and clustering by 3D Skeletonization *(ongoing)* | Python, Mathematica — *12/2022 ~ Present*

Supervised by Prof. Tao Ju, Washington University in St. Louis

- Used spatial density data as input, graphed the feature changes to automatically search for a density threshold to binarize the input, and output more than 50 features after computing skeletons. Finally use these features to cluster data and find out the target genotype as a conclusion.

3D Scene Rendering Based on Ray Tracing and Photon Mapping | C++ — *07/2020 ~ 09/2020*

Guided by Prof. James O' Brien, University of California, Berkeley

- Constructed a ray tracer to render images. That is, computing the visibility of objects then shading it by Phong model. Defined a ray starting from the eye to the scene to render the whole scene with specular reflection computation. Extended the method to use Photon Mapping.

Mathematical Contest in Modeling (CUMCM) | Matlab — *09/2019*

- Participated in a mathematical modeling research topic, that is, to use a drum to bounce a ball and analyze the strategy in different conditions. Proposed a model to compute mechanical data. Used the space coordinate series to make the equations and used the mathematics software to solve them. Completed a thesis and won the national 2nd prize, which is in the top 3.8% of all competitors.