Chaos Paddling: A New Vision on Rapids

Blake Craig

Chaos Paddling is a potential business idea that I have, and using this class, I will be able to make the first steps. The goal is to provide real-time viewing of paths that a paddler can take down a rapid. For this class, I want to overlay a route on a luminance image that is uploaded. If I am making a lot of progress, potentially creating a 3D model of this route and the rapid is an option.

I plan on using Dijkstra's algorithm to find the shortest path between frothy bits of water, indicating the flow throughout the rapid. The middle of these frothy bits will be attached, creating the route itself. Using a colored line, a route is displayed for the user to take down a rapid.

The main problem that I have with data collection and my algorithm is route detection behind an item. In some instances, a route may be hidden behind a rock or drop lower than a certain angle can capture, leading to the unknown of the rapid. In the final product, I want to be able to capture the route, even if it requires going behind objects. Thus, I need to add much to the algorithm to determine where a route goes and where it continues. This will be extremely hard as water is not stagnant and can change very rapidly.

This project will take on feature mapping and image manipulation, core components of image processing. I haven’t fully fleshed out the strategies or algorithm to find the best route or edit the image to good standards, but there are many components from contrast to Gaussian blurs that contribute to the final product.

My goal is to learn different strategies to produce the route that will be efficient and effective. Additionally, I hope to gain insight before I add AI, so then the AI is simply a supplemental source for the project. My goal is to not rely on AI for the final product, therefore, allowing an offline experience can still be available to customers. I hope to dive deeper into object detection and edge detection, differentiating objects from water. For future initiatives, I hope to add more water physics, like hydraulics and general flow.

**March 18 (Week 1)**

* Finalize project scope and core objectives
* Develop initial proposal draft
* Plan technical approach (image processing methods, graph-based routing, visualization)

**March 25 (Week 2) – Project Check #1 (March 24)**

* Submit Project Proposal (March 19)
* Implement basic obstacle detection (distinguish rocks from water)
* Begin water flow detection (gradients + variance methods)
* Run test cases on sample images

**April 1 (Week 3)**

* Refine water flow detection (improve turbulent water vs. still water)
* Implement river angle estimation for better route calculation
* Optimize graph-based shortest path algorithm
* Run tests with multiple river images

**April 8 (Week 4)**

* Improve route visualization (refine overlays for obstacles, water, and path)
* Adjust route weighting system (flow-heavy vs. shortest path balancing)
* Debug false positive obstacles in rapids

**April 15 (Week 5) – Poster Draft Submission (April 16)**

* Refine and finalize main algorithm (object detection, flow analysis, route computation)
* Submit Poster Draft
* Generate sample visualizations for poster

**April 22 (Week 6) – Project Check #2 (April 21)**

* Submit final poster design to Davidson printer (April 22)
* Optimize performance (speed, memory usage, scalability)
* Conduct multiple test runs on different river images
* Improve user interface/output presentation

**April 29 (Week 7) – Project Check #3 (April 23)**

* Conduct final debugging and ensure all features work as expected
* Prepare 15-minute class presentation (April 30)
* Run a real-world simulation or test case

**May 6 (Week 8) – Presentation Week**

* Present 15-minute project demo to class (April 30 or May 5)
* Final tweaks based on class feedback
* Prepare for Verna Miller Case Symposium (May 8)

1. **"ROSEBUD: A Deep Fluvial Segmentation Dataset for Monocular Obstacle Detection in USV Navigation"**  
   <https://pmc.ncbi.nlm.nih.gov/articles/PMC9269472/>
2. **"Autonomous Navigation and Collision Prediction of Port Channel Traffic Based on AIS Data"**  
   <https://www.nature.com/articles/s41598-024-60327-9>
3. **"Progress Towards Multidimensionally Scalable Assisted and Autonomous Vessel Navigation"**  
   <https://www.tandfonline.com/doi/full/10.1080/20464177.2025.2476862>
4. **"River Mapping from a Flying Robot: State Estimation, River Detection, and Obstacle Mapping"**  
   <https://www.researchgate.net/publication/257523114_River_Mapping_From_a_Flying_Robot_State_Estimation_River_Detection_and_Obstacle_Mapping>
5. **"Obstacle Detection and Safely Navigate the Autonomous Vehicle Using Deep Neural Network"**  
   <https://pmc.ncbi.nlm.nih.gov/articles/PMC7506726/>
6. **"Deep Learning for Autonomous Ship-Oriented Small Ship Detection"**  
   <https://www.sciencedirect.com/science/article/abs/pii/S0925753520302095>
7. **"Path Following, Obstacle Detection and Obstacle Avoidance for Underwater Snake Robots"**  
   <https://www.frontiersin.org/articles/10.3389/frobt.2019.00057/full>
8. **"Real-Time Obstacle Detection for Unmanned Surface Vehicle Using Deep Learning"**  
   <https://pubs2.ascee.org/index.php/IJRCS/article/view/1147>
9. **"Nonlinear Model Predictive Control with Obstacle Avoidance Constraints for Autonomous Navigation in a Canal Environment"**  
   <https://arxiv.org/abs/2307.09845>
10. **"Temporal Context for Robust Maritime Obstacle Detection"**  
    <https://arxiv.org/abs/2203.05352>
11. **"Research on Detection of Floating Objects in River and Lake Based on AI Intelligent Image Recognition"**  
    <https://arxiv.org/abs/2404.06883>
12. **"Velocity Obstacle"**  
    <https://en.wikipedia.org/wiki/Velocity_obstacle>