

## **NapCAD Project Proposal**

**Katie Hite, Kiki Chandra, Caleb Kissel, and Celine Ta**  
**Software Design, Spring 2015**

- 1. The Big Idea: What is the main idea of your project? What topics will you explore and what will you generate? What is your minimum viable product? What is a stretch goal?**

NapCAD is an openCV-based image to CAD program that translates JPEG images of “napkin sketches” (orthographic views) to STL files. We’ll explore openCV pathfinding, 3D geometries from 2D points, and external STL packages.

Specific data types TBD.

MVP: Read in 6 drawings on different views of object, scanned in specific order of views. Create an STL file of cube with those 6 faces. Any planned extrusions/ holes are drawn on the face, but are not actually extruded. Expectation is to only use rectangles in this version.

Extension 1: Add a GUI.

Extension 2: Add extrusions

Extension 3: Add ability to make holes with dotted lines.

Extension !: Add ability to CAD organic shapes.

- 2. Learning Goals: Since this is a team project, you may want to articulate both shared and individual learning goals.**

Team--

- Work effectively as a relatively large (4 person) team
- Use git for proper version control
- OpenCV experience
- Integration of multiple, clearly delineated components

Individual--

Celine

- Connecting user interface to program
- Current user interface/ aesthetics standards (i.e. Apple, etc.)/ creating information flows, etc.
- Connecting multiple APIs
- Program design (i.e. how to best structure classes, etc.)
- Algorithms, if applicable, especially efficiency and learning

Kiki:

- Creating an aesthetically pleasing and efficient GUI
- Developing a mathematical system to use lengths taken from images and creating a 3D figure (optimization of algorithms?)
- Become more proficient in using OpenCV

Katie:

- Creating simple and easy to use interfaces for programs
- Using OpenCV in a more efficient/real time way
- Using readable program flows and providing better documentation

Caleb:

- Gain experience with OpenCV
- Learn about CAD automation
- Improve pair programming skills

- 3. Implementation Plan: this will probably be pretty vague initially. Perhaps at this early juncture you will have identified a library or a framework that you think will be useful for your project. If you don't have any idea how you will implement your project, provide a rough plan for how you will determine this information.**

We have identified potential libraries/applications that we can use to convert our readings from OpenCV to .csv files, and then to .stl.

- Pathfinding: Convert image to black/white. Use openCV's cv2.findContours function to find paths
- 3D Mapping: Scale to desired size (based off reference distance). Output array/mesh/something of 3D coordinates (vertices for MVP, may change for future extensions) - either represent shapes as matrices and work with transforms/rotation/etc. or create sets of vertices for the 3D shape by pulling separate measurements from each drawing part?
- STL Conversion: Output to STL
  - Blender, PythonOCC, PythonSTL, FreeCAD Python API

- 4. Project schedule: You have 6 weeks (roughly) to finish the project. Sketch out a rough schedule for completing the project. Depending on your project, you may be able to do this in great specificity or you may only be able to give a broad outline. Additionally, longer projects come with increased uncertainty, and this schedule will likely need to be refined along the way.**

<b>Week 1</b>	<b>3/22 - 3/28</b>	<b>Project Proposal, Research (all packages/ libraries specced downloaded), Minimal tests of each program part (pathfinding, mapping, conversion)</b>
<b>Week 2</b>	<b>3/29 - 4/4</b>	<b>Design Review (4/2) + Framing/ Agenda-setting document, Parts of program functioning independently</b>
<b>Week 3</b>	<b>4/5 - 4/11</b>	<b>MVP (Integrated), Design Review Reflection (4/6)</b>
<b>Week 4</b>	<b>4/12 - 4/18</b>	<b>Start Project Website Code Review document and Code Review (4/13), Code Review Reflection (4/16), Refine MVP integration as needed Start Extension 1 (if time allows)</b>
<b>Week 5</b>	<b>4/19 - 4/25</b>	<b>Extension 1 Start Extension 2/3</b>

<b>Week 6</b>	<b>4/26 - 4/30</b>	<b>Continue with Extensions</b>
<b>Finals</b>	<b>5/6</b>	<b>Expo-style poster/ Demo</b>

- 5. Collaboration plan: How do you plan to collaborate with your teammates on this project? Will you split tasks up, complete them independently, and then integrate? Will you pair program the entire thing? Make sure to articulate your plan for successfully working together as a team. This might also include information about any software development methodologies you plan to use (e.g. [agile development](#)). Make sure to make clear why you are choosing this particular organizational structure.**

Because we are all interested in multiple aspects of the project, we plan to mainly pair-program. We have not decided how to break into groups yet but will probably try to organize them such that everyone will get to work at least a bit on all the parts that align with their individual goals. Each part of the program will be developed concurrently.

- 6. Risks: What do you view as the biggest risks to the success of this project?**

There are several potential stumbling blocks for this project. Though we have several group members who have used OpenCV before, there are always complications. The image processing must be finished before we can do anything else, and we must make sure not to get bogged down and spend too much time working only on this.

In addition, the actual interpretation will be very difficult. This will need to be a fairly intelligent and modifiable algorithm, and could prove very difficult to write. This will likely take a great deal of time, and should be started early.

Finally, interfacing with a library or another program has many challenges, and we need to make sure we have some understanding of the CAD library we choose early on.

Hardware: We plan to use our laptop webcams, but we may end up using an external webcam if we face difficulties in standardizing light and angle of the laptop webcam.

- 7. Additional Course Content: What are some topics that we might cover in class that you think would be especially helpful for your project?**

- Potential challenges with integrating multiple APIs?
- Basic algorithm design principles and things to keep in mind (Big O time important/not important, etc.)