Project Report: VideOrigami

Howie Wang

Link to the deployed webpage: https://whyhowie.github.io/videorigami.

(Note: An error will appear in the console regarding the `allow-presentation` flag. I suspect that this is an error that occurs when a `Player` object is created through the YouTube Player API. Any attempt to add the flag afterwards will not affect the object initialization and thus will not fix the error. This error does not affect the user experience and can be ignored.)

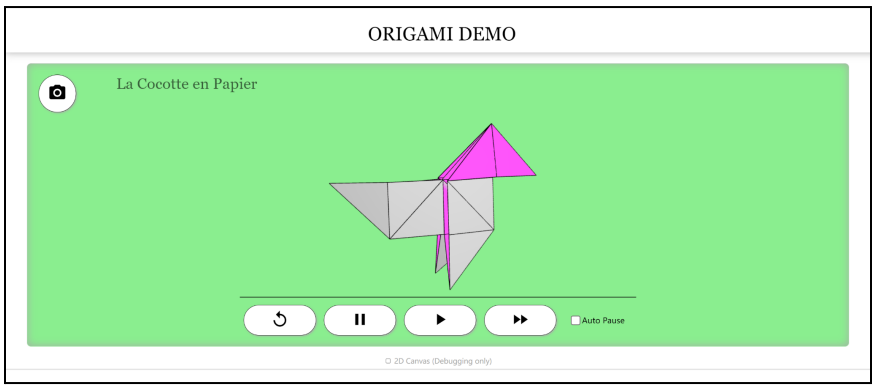


Figure 1. 3D visualization of an origami model presented in the web application.

Project Description

The purpose of this web application prototype (partially shown in Figure 1) is to demonstrate alternative 3D views of beginner origami models.

For many new origami hobbyists, following step-by-step origami diagrams and watching online video tutorials are two convenient ways to learn how to fold origami models. However, certain diagrams in origami instructions can be confusing, as 2D graphics may not effectively represent certain folds. In the meantime, the quality of online video tutorials can vary. Sometimes the lighting or the contrast of the origami paper used for the video is not ideal. The presenter's hands may also occlude the view. Therefore, having additional visual aids to supplement the 2D diagrams and videos may be helpful. This project will be a demonstration of the concept, presenting one origami model step-by-step through interactive 3D animation.

The bottom half of the webpage includes a video with diagrams serving as clickable timestamps for the video. The viewer may jump to the corresponding moments in the video to learn about the folds of interest.

This web application can be engaging as it combines 2D content with 3D and static elements with animation. It also offers large buttons and bright, contrasting colors, making it accessible to people with color blindness.

Interactive Features

The current prototype includes the following interactive features:

- Responsive design (viewable on different screen sizes)
- 3D animation window of the origami model:
    - Upon loading, the animation will start automatically;
    - Using a mouse, drag while holding the left mouse button to rotate the view; drag while holding the right mouse button to pan; scroll up and down to zoom in and out;
    - Alternatively, on a touch screen, drag with one finger to rotate the view; drag with two fingers to pan; pinch in and out to zoom in and out;
    - The four large buttons correspond to restarting, pausing, continuing, and fast-forwarding the animation; check the Auto Pause checkbox to allow the animation to pause at each step;
    - Click on the Camera button located in the top left corner of the 3D animation window to take a screenshot - this will open a new tab showing the screenshot with a transparent background.
- Interactive YouTube video tutorial with corresponding timestamps:
    - Scroll down to the second half of the page to view the video tutorial and the step-by-step diagrams; (Note that this prototype does not feature the standard diagrams indicating types of folds. Screenshots of the 3D model are used instead. These can be replaced by some high-quality vector diagrams if time permits.)
    - Using a mouse, while hovering the cursor in the diagram area, scroll horizontally to browse the diagrams; click on one of the diagrams to jump to the corresponding part of the video - the diagram will be highlighted and centered;
    - On a touch screen, swipe across the diagram area to browse the diagrams; tap on one diagram to view the linked video moment;
    - While the video is playing, the corresponding diagram at each timestamp will also be automatically highlighted and centered.

Tools Used

This web application is developed using mostly basic HTML, CSS, and Javascript. The following external tools and libraries are used to achieve the functionality and visual effects:

- OriSim3D by Rémi Koutcherawy. The repository contains command-based algorithms to generate and animate origami models in the form of 3D geometries. It is one of the few

available applications that visualize and animate origami folds step by step. The model and animations can be defined using lines of text. The algorithm is fundamental to this web application. The code was reorganized into three files and imported as ES6 modules.

- Three.js library. The 3D geometries mentioned above are rendered directly through WebGL, providing limited interactive functionalities. With the Three.js library module and add-ons, coloring and shading can be defined more easily and interactive view manipulation (OrbitControls) is also allowed. It is imported as an ES6 module.
- YouTube Player API. The API is used for embedding the YouTube-hosted video tutorial. It allows most features available to locally hosted videos without needing extra storage for the video. It is included in the HTML file using the `<script>` tag.
- Material Symbols and Icons by Google. The icons are used on the buttons to achieve a consistent, user-friendly look.

Design Iterations

Most of the design changes was made after the FP2 lab session. Based on the peer feedback, the timestamp was changed into the horizontal arrangement and the user could either choose to view the 3D model or watch the video tutorial along with the diagrams.

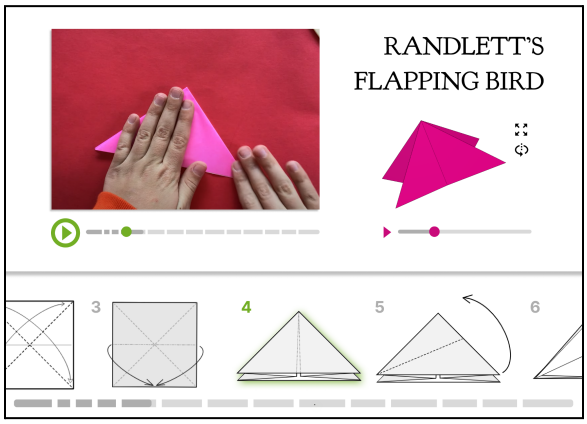


Figure 2. Revised design for Prototype 1.

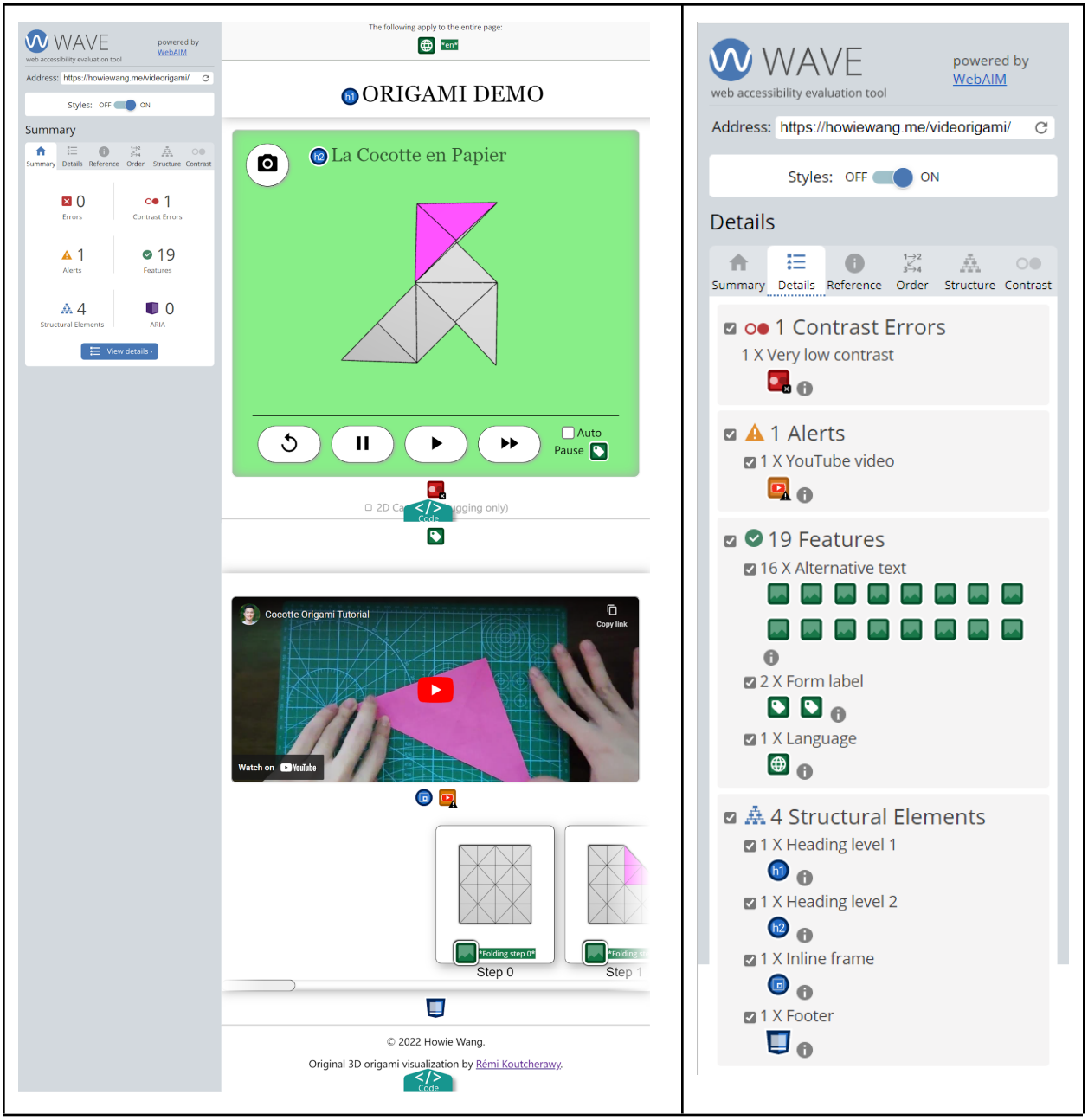
Challenges

The actual implementation is slightly different from the proposed one (Figure 2), in that I ran into technical difficulties in combining React with the OriSim3D codebase.

I spent the majority of my time studying algorithms for 3D origami visualization. I also learned Three.js without prior knowledge.

Appendix: Accessibility Check

Wave was used to evaluate the accessibility of the prototype. One contrast error was detected but it was due to a clickable checkbox to show and hide the 2D crease pattern. It is a semi-hidden feature for debugging and testing command lines for origami animation. This error will not affect the user experience and can be ignored.



(a) (b)

Figure A1: Accessibility check on Wave.