Interactive Art Installation PATHS

Part 1: Introduction

PATHS is an interactive art installation that incorporates p5.js and a face-tracking algorithm OpenCV to create an immersive experience for the audience. In this project, the basic idea is to simulate a visual metaphor where words like "slut," "whore," and "cheap" cascade from the top of the screen and come to a halt when they touch the viewer's face, evoking a powerful sense of metaphorical humiliation and mimicking the experience faced by women in their everyday life. This design will be interactive and engaging by allowing viewers to actively participate in the experience, taking them on a journey from the initial cascade of derogatory words to a moment of reflection and dialogue as they interact with the installation.

Therefore, the purpose of PATHS is to challenge societal norms and confront the issue of slut-shaming in today's society. I wish PATHS could encourage reflection and dialogue on the issues of slut-shaming, fostering engagement and prompting visitors to contribute to the ongoing conversation. The minimalist visual style enhances the impact, emphasizing the powerful message. The target audience includes individuals interested in societal issues, gender equality advocates, and everyone seeking an immersive and thought-provoking interactive art experience.

Part 2: User Interactions

- Move Around the Head to Test Face Tracking:
  The user's facial movements will trigger face tracking. The user needs to grant camera access to the webpage (Chrome preferably) so that the face-tracking algorithm can detect their face. And once they see the green dot on the camera (indicating camera is working properly), they can move around their face to test the face-tracking feature. The algorithm detects and responds to these movements in real-time, enhancing the interactive experience. It might take some time for the algorithm to load on some users' computer. The user's face will be outlined with white lines.
- Visual Engagement with Falling Words:
  Humiliating words will cascade from the top of the screen, halt when they touch the viewer's face, and pile up when they stop falling to the bottom of the screen. Participants can move their face around to see if the words will stop falling when they touch the face, which creates a powerful and immersive experience.

Part 3: External Tools Used

- p5.js
  Why: the choice of p5.js is driven by its simplicity and versatility in creating interactive visual graphics. It aligns seamlessly with the project's goal of providing an engaging and visually appealing experience.
  How: I incorporated the OpenCV in p5.js and since the original face-tracking algorithm displays the user's real face and background, I utilized p5.js to paint the background black and only keep the outline of the user's face. In this way, it preserves a better artistic style than displaying the user's real face and background directly.

What it Adds: The integration of p5.js shifts the practical use of the original algorithm to a more artistic use and it maintains the minimalistic visual style. Also, it adds a layer of interactivity to the project, allowing for the creation of visually compelling graphics.

- OpenCV
  Why: the selection of OpenCV as the face-tracking algorithm was driven by the need to incorporate accurate facial recognition into the interactive experience. OpenCV is a robust and widely used tool for computer vision tasks.
  How: I used OpenCV's algorithm to detect the participant's face and I found the p5.js version of OpenCV but was not working smoothly. So I broke down the algorithm into parts and removed some detection points on the face so it would work more smoothly.
  What it Adds: the use of OpenCV builds the foundation of the interactive aspect of the installation. It introduces a dynamic element by accurately tracking the viewer's face, allowing the falling words to respond in real time to the user's movements. This dynamic interaction enhances the overall impact of the visual metaphor and contributes to the personalized nature of the experience. The incorporation of OpenCV aligns with the project's aim to provoke thought and reflection on societal issues through innovative technological solutions.

Part 4: Prototypes Iteration
During the development of PATHS, user testing played a pivotal role in shaping the project's visual design. Two prototypes were created, each with distinct visual styles. Prototype 1, featuring a minimalist design, garnered positive feedback for effectively conveying the falling words and engaging users. In contrast, users found Prototype 2's pixelated style confusing and less visually attractive. The feedback guided the decision to choose Prototype 1 as the final design.
Simultaneously, the integration of OpenCV into the p5.js environment presented challenges. To incorporate OpenCV into p5.js, I had to do extensive research and see how other developers were implementing it. The first version of the algorithm seemed to be slow and buggy on some users' computers during my user testing session. Therefore, I had to break down the code and by experimenting with adding and removing some detection points on the face, I improved the algorithm in multiple iterations.

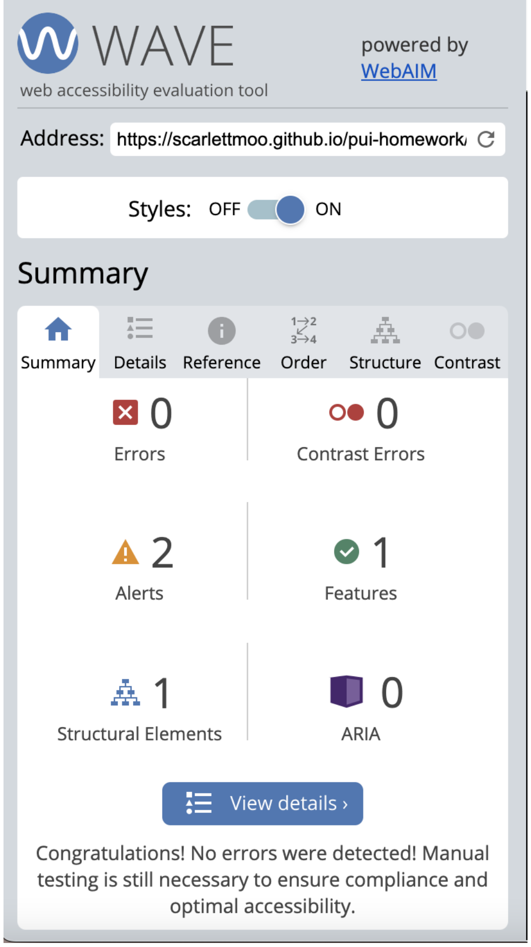
Part 5: Challenges Encountered
The implementation of PATHS encountered challenges in optimizing loading times, particularly concerning the face-tracking functionality. At first, it took a long time for face tracking to load and it stumbled a lot throughout the experience in my early user testings. Additionally, it took me a long time to determine the artistic visual design of the page since the original OpenCV code directly shows the real-time face and background. Thus, I had to design two prototypes and I was glad the minimalist one had good feedback and had a great way of conveying the message.

Appendix

The website is tested on iPads, Laptop, and iPhones.

Deployed Link:

https://scarlettmoo.github.io/pui-homework/final-project/final-project/final-project/index.html



WAVE web accessibility evaluation tool · powered by WebAIM

Address: https://scarlettmoo.github.io/pui-homework/

Styles: OFF ● ON

Summary

☒ 0 Errors

0 Contrast Errors

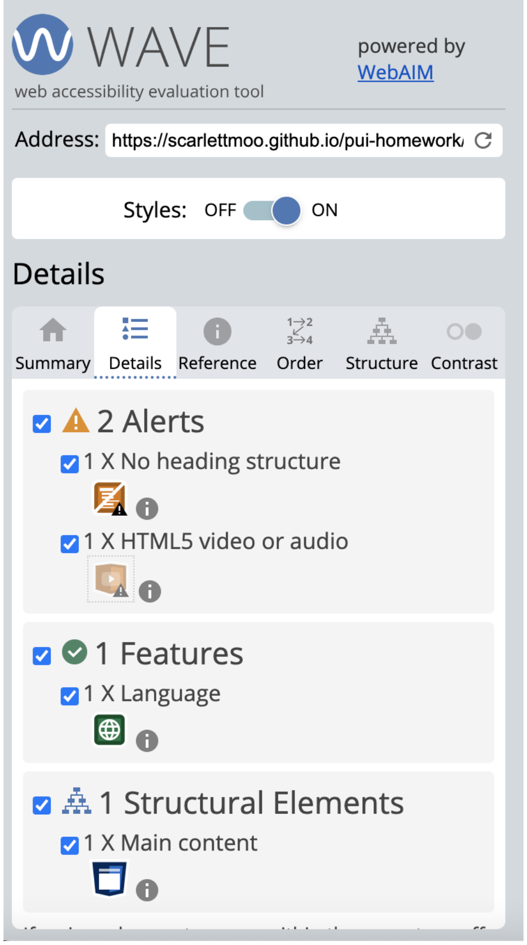
⚠ 2 Alerts

✓ 1 Features

1 Structural Elements

0 ARIA

Congratulations! No errors were detected! Manual testing is still necessary to ensure compliance and optimal accessibility.



WAVE web accessibility evaluation tool · powered by WebAIM

Address: https://scarlettmoo.github.io/pui-homework/

Styles: OFF ● ON

Details

☑ ⚠ 2 Alerts

- ☑ 1 X No heading structure
- ☑ 1 X HTML5 video or audio

☑ ✓ 1 Features

- ☑ 1 X Language

☑ 1 Structural Elements

- ☑ 1 X Main content