

Thesis Proposal:

Accessible geoprocessing in the browser using
WebAssembly & Visual Programming

Msc Geomatics for the Built Environment

Thesis Proposal : Accessible geoprocessing in the browser using WebAssembly & Visual Programming

MSc Geomatics for the Built Environment

January 28, 2022 @ TU Delft Faculty of Architecture, Room U

Proposer: Jos Feenstra

First supervisor : Stelios Vitalis

Second supervisor : Ken Arroyo Ohori

External Examiner : André Mulder

Chapters:

1. Introduction

2. This Study

3. Methodology

4. Preliminary Results

1. Introduction

Geographical Web Applications

- Type of GIS
- Important Component of Geo Informatics
- Often first and only experience of GIS for the general public.
- Extremely accessible
- Easy to update & maintain

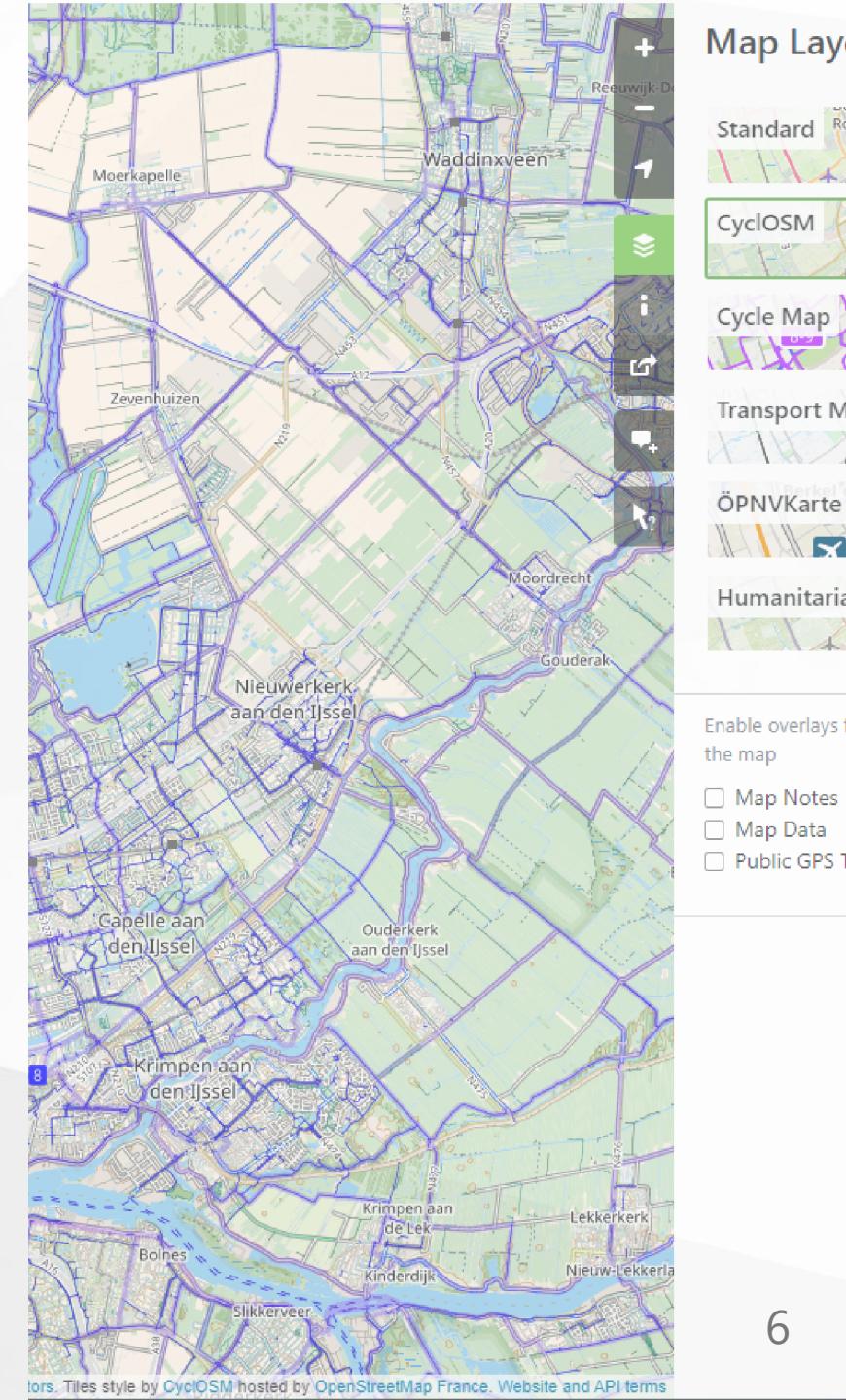


Geographical Web Applications

Examples

- Google Maps
- Open Street Map
- PDOK viewer

Focus: *Looking at geodata*



Next: *Using Geodata*

- Who? Data Users
 - Scientists, analysts, architects, urban planners, municipalities
- Simple queries
- Advanced queries
- Involves Processing & analyzing geodata



Possible, but not using geo web apps...

- Only viewers
- Very limited Geoprocessing or analysis
- Data users are left to their own devices.
 - Make jump to specialist GIS software
 - Use something known (CAD / modelling)
 - ~~Tracing Google Maps Screenshots~~



The Idea: Geodata Processing *within* a browser

Browser-based Geoprocessing

- or Client-side Geoprocessing (CSG)
- Relatively unknown, but gaining traction.
- Potential of accessible and sharable processing tools

"But, why is client-side geoprocessing as of yet still nowhere to be found?"

Obstacles

1. CSG is technically challenging
2. CSG is underdeveloped
3. CSG is overshadowed by other technologies

Connected Obstacles

1. CSG is technically challenging
2. CSG is underdeveloped
3. CSG is overshadowed by other technologies

2. This Study

Aim

- *Actualizing* client-side geoprocessing
- Methodology:
 - Practical
 - Wholistic
 - Precise

First obstacle : CSG is a technical challenge

- JavaScript is not performant
- No mature geoprocessing libraries available client-side

Methodology : Research if WebAssembly can be used to compile existing geoprocessing libraries.

Second Obstacle : CSG is immature

- Little to no supporting software
- no noteworthy examples

Methodology : Develop a use-case application to support the research and to serve as example for future research.

Third Obstacle : CSG is considered unnecessary

- overshadowed by native & server side geoprocessing

Methodology : Develop a use-case application to demonstrate the situational advantages of client-side geoprocessing

Central Question:

"How to design and create a browser-based GIS environment which can effectively utilize existing geoprocessing libraries, using only the current state of standard client-side technologies"

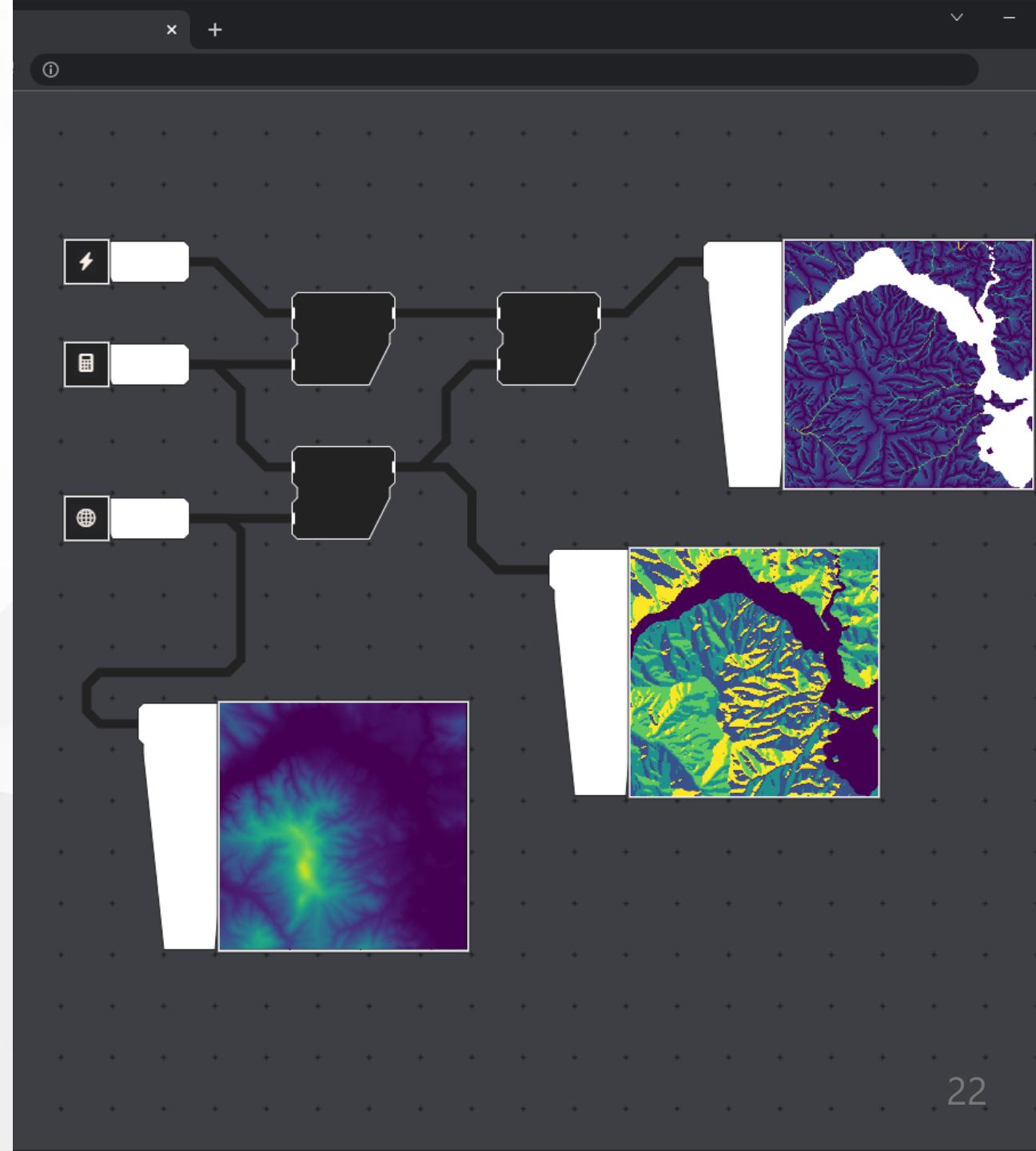
*"How to design and create a browser-based GIS environment which can **effectively utilize** existing geoprocessing libraries, using only the current state of **standard client-side technologies**"*

Sub Questions

1. *What is the most fitting methodology of compiling C++ geoprocessing libraries to Web-Assembly?*
2. *How to design and create a client-side geoprocessing environment for data-users?*
3. *How can wasm-compiled geoprocessing libraries be distributed and used in a client-side geoprocessing environment?*
4. *What are the advantages and disadvantages of GIS applications created using a client-side geoprocessing environment powered by WebAssembly?*

Use Case : GeoFront

- Web-first GIS
- Offers full geodata analysis procedure
- Visual Programming Language



3. Methodology

Methodology

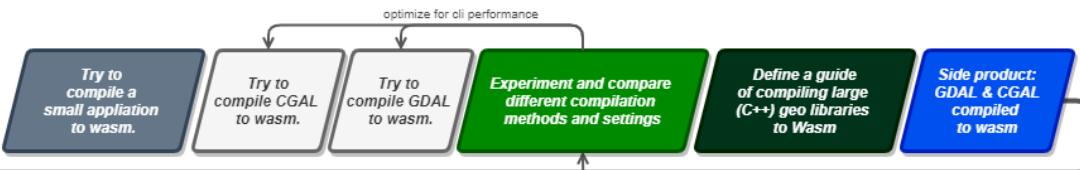
- Incremental
- Iterative

Per phase:

- Answer sub question
- Develop use-case component

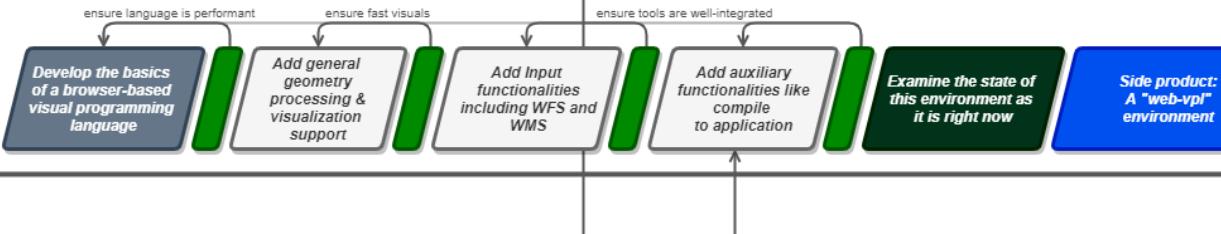
Phase 1. Compile

What is the most fitting methodology of compiling C++ geoprocessing libraries to WebAssembly?



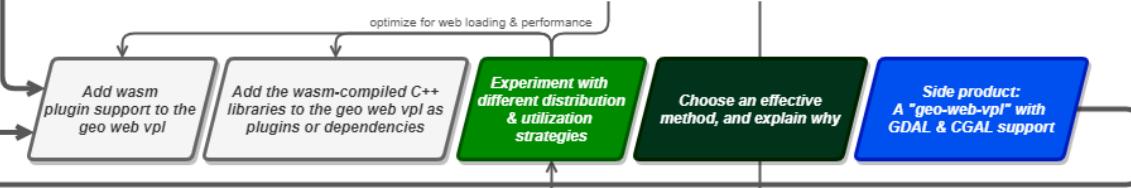
Phase 2. Interface

How to design and create a client-side geoprocessing interface for data-users?



Phase 3. Distribute

How can wasm-compiled geoprocessing libraries be distributed and used in a client-side geoprocessing interface?



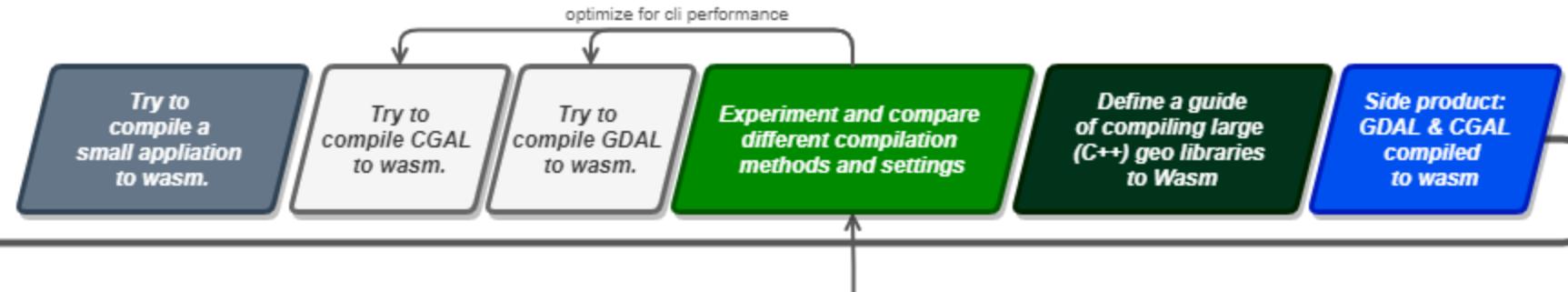
Phase 4. Utilize

What are the advantages and disadvantages of GIS applications created using a client-side geoprocessing environment powered by WebAssembly?



Phase 1. Compile

What is the most fitting methodology of compiling C++ geoprocessing libraries to WebAssembly?

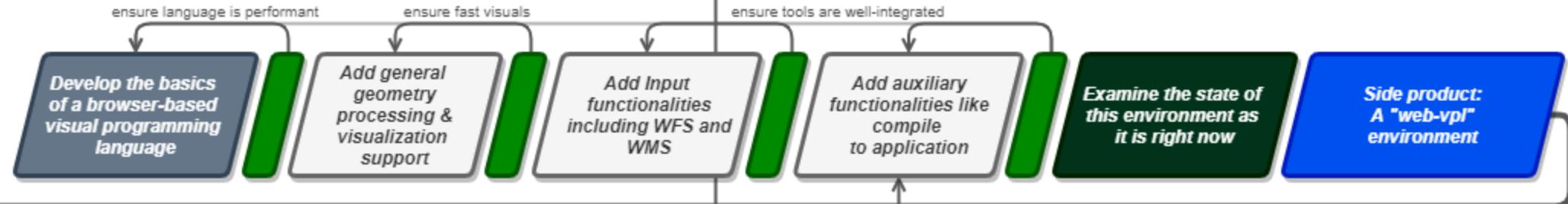


Phase 1

- Study WebAssembly capabilities & features
- Compile C++ libraries to Wasm (GCAL / GDAL)
- Analyse and compare performance.

Phase 2. Interface

How to design and create a client-side geoprocessing interface for data-users?



Phase 2

- Study existing visual geometry processing languages
- Design & develop a visual language
- Add necessary features step by step

Phase 3. Distribute

How can wasm-compiled geoprocessing libraries be distributed and used in a client-side geoprocessing interface?

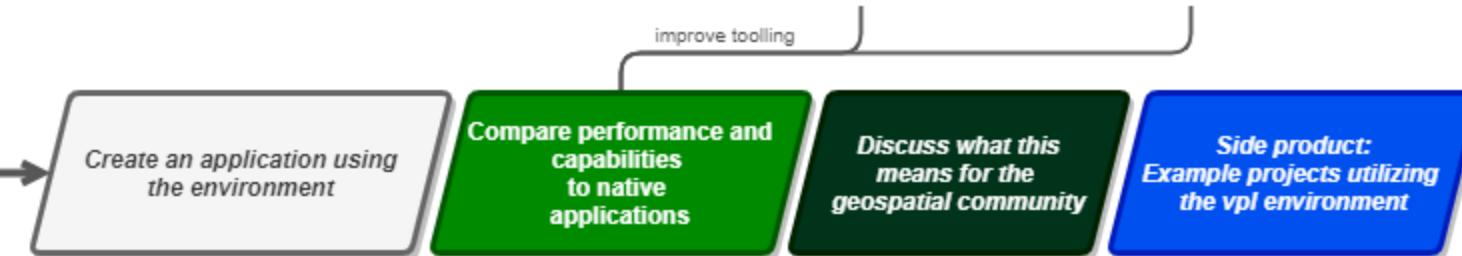


Phase 3

- Study code splitting & distribution methods
- Combine and align the results of phase 1 & 2
- Reassess the wasm-compiled libraries

Phase 4. Utilize

What are the advantages and disadvantages of GIS applications created using a client-side geoprocessing environment powered by WebAssembly?



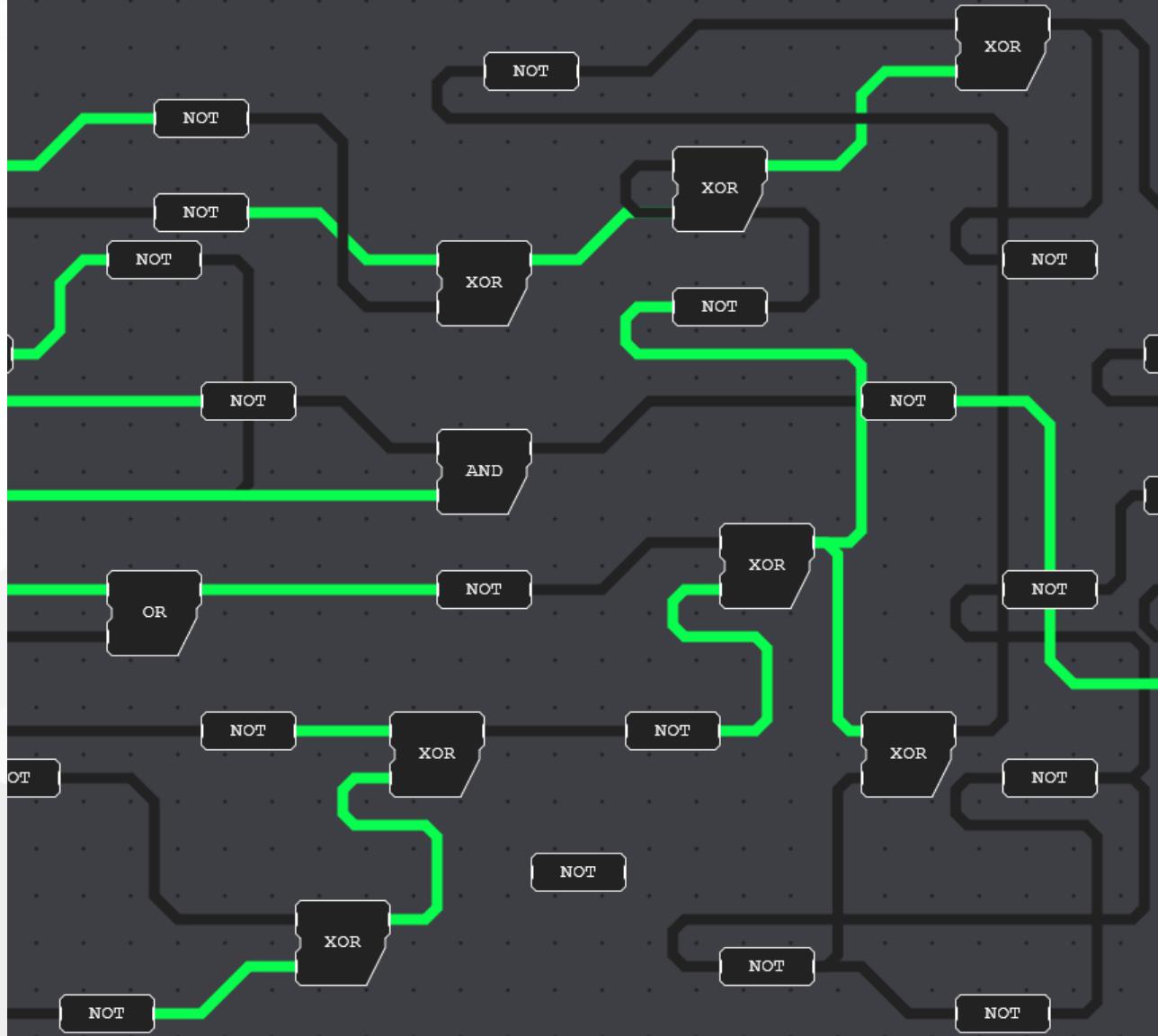
Phase 4

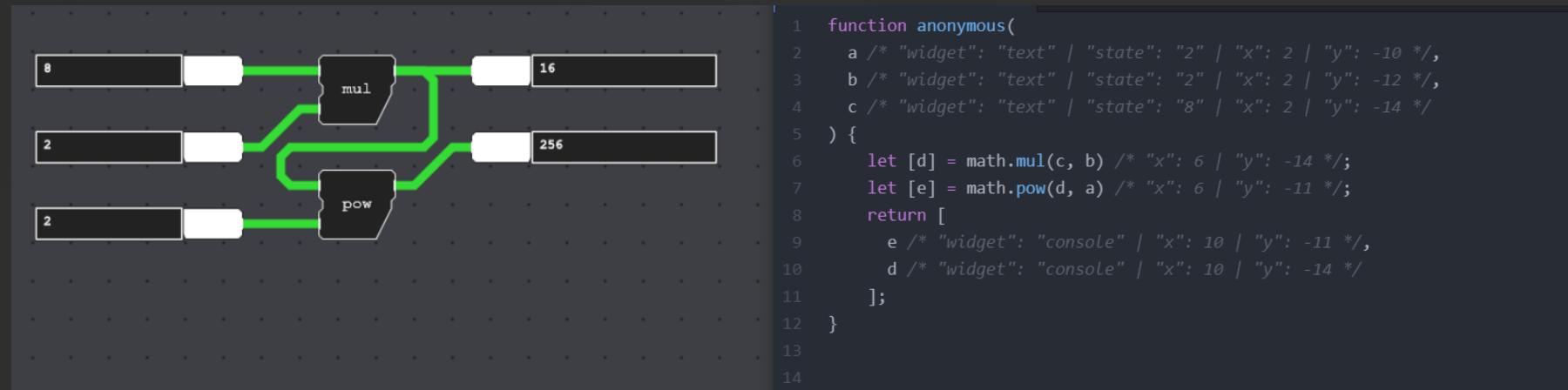
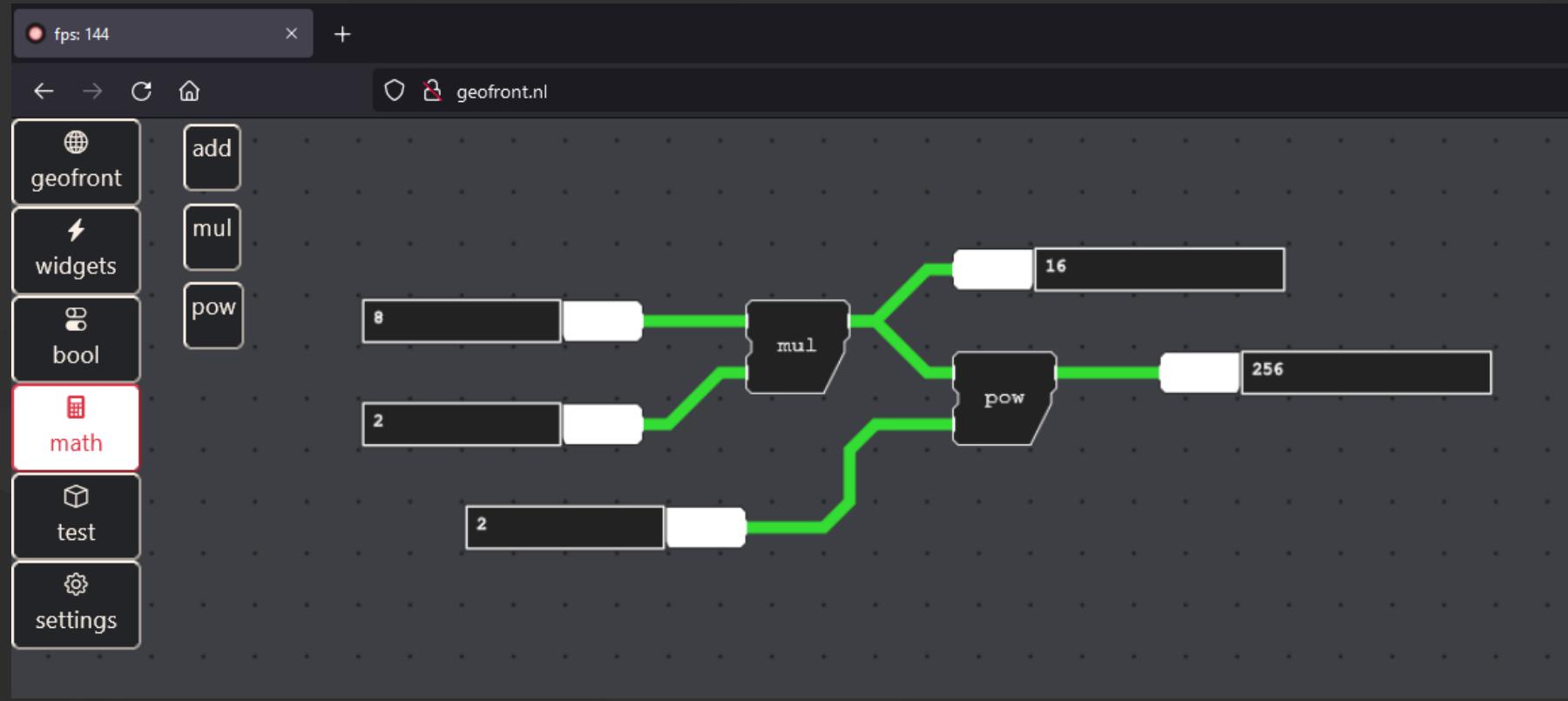
- Study the environment itself
- Use it to develop geoprocessing applications
- Assess its performance and usability

4. Preliminary Results

1. GeoFront

- First draft of the use-case application
- geofront.nl
- Focus: basics of visual programming





2. WebAssembly

- Research assignment
- Cityjson Validator
- Rust & Wasm



(cjval v0.3.0 is used)

(files are never uploaded, validation is done locally)

The file is 100% valid!

Good idea to upgrade to v1.1

details

tency

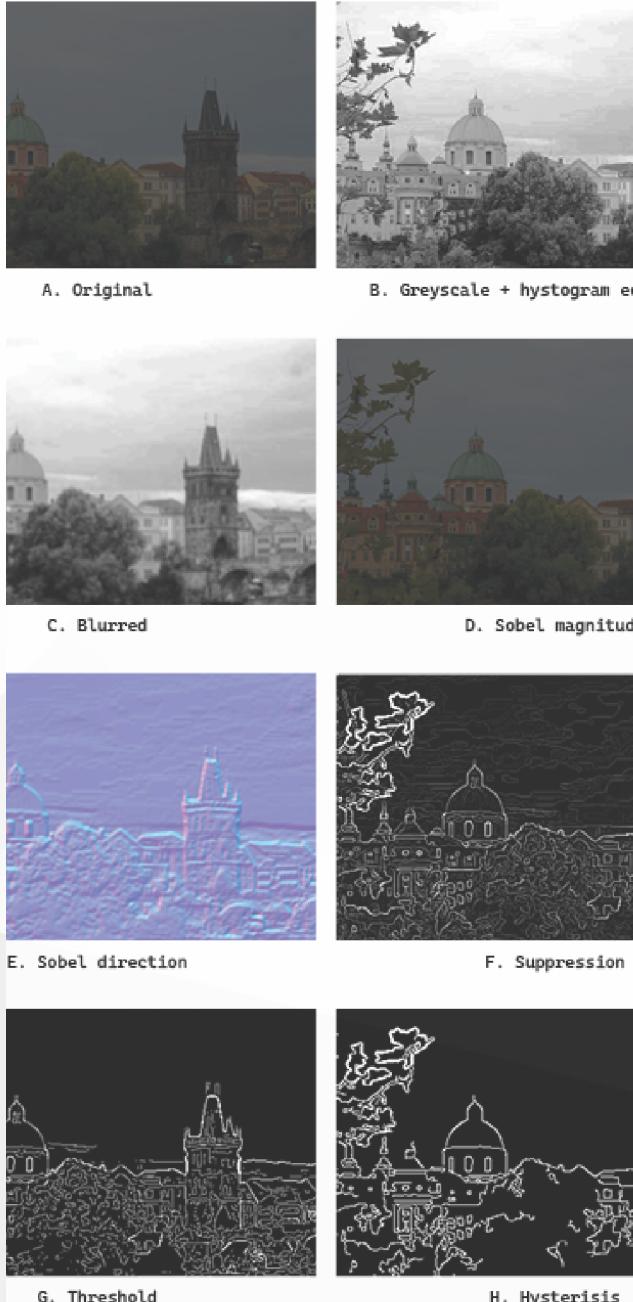
ings)

warnings)

gs)

3. Internship

- Web (Geo) Data processing
- Image processing
- Commercial perspective



1/4

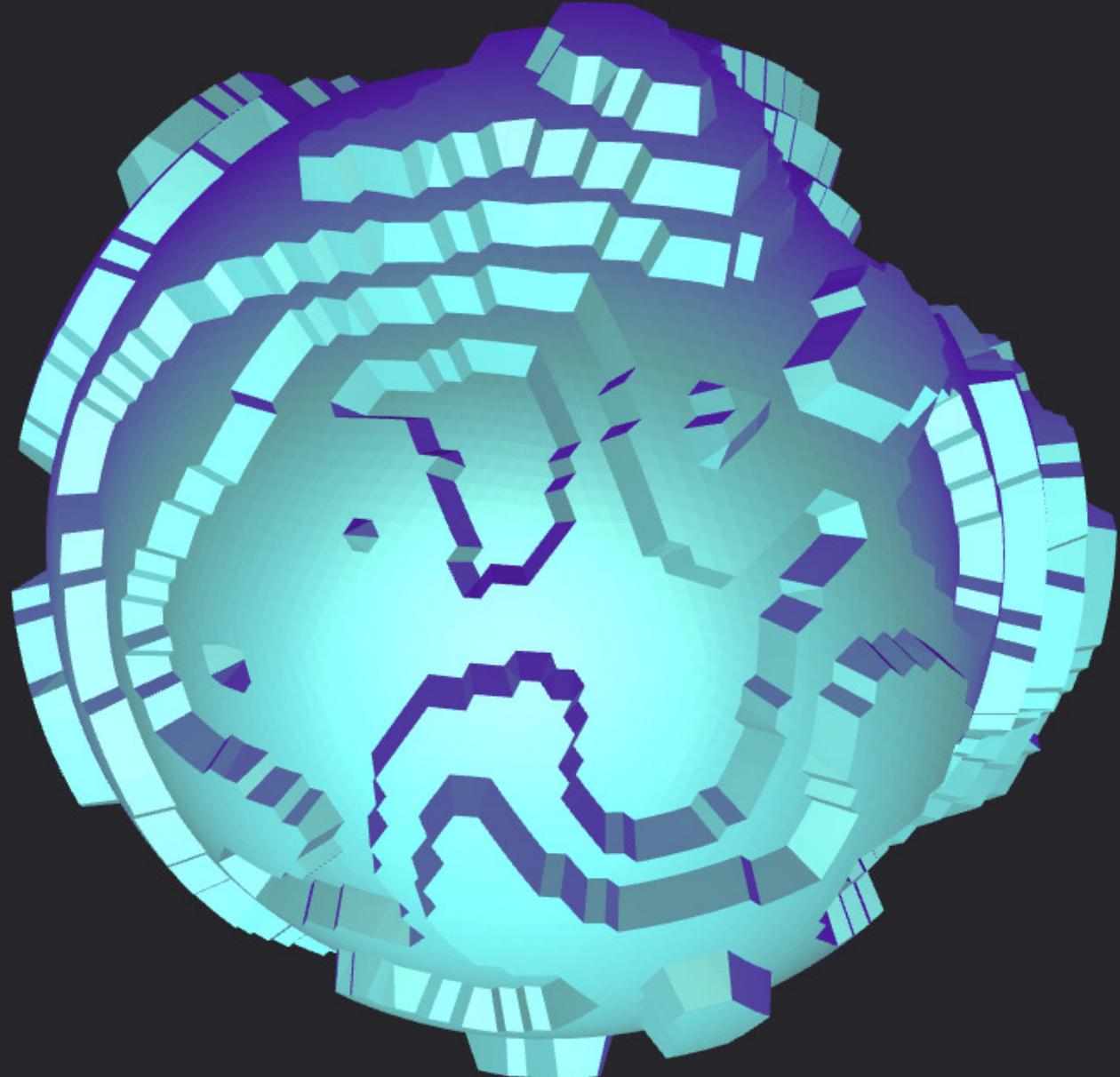
Please Load a scan

↻ Reset

↑↓ Submit

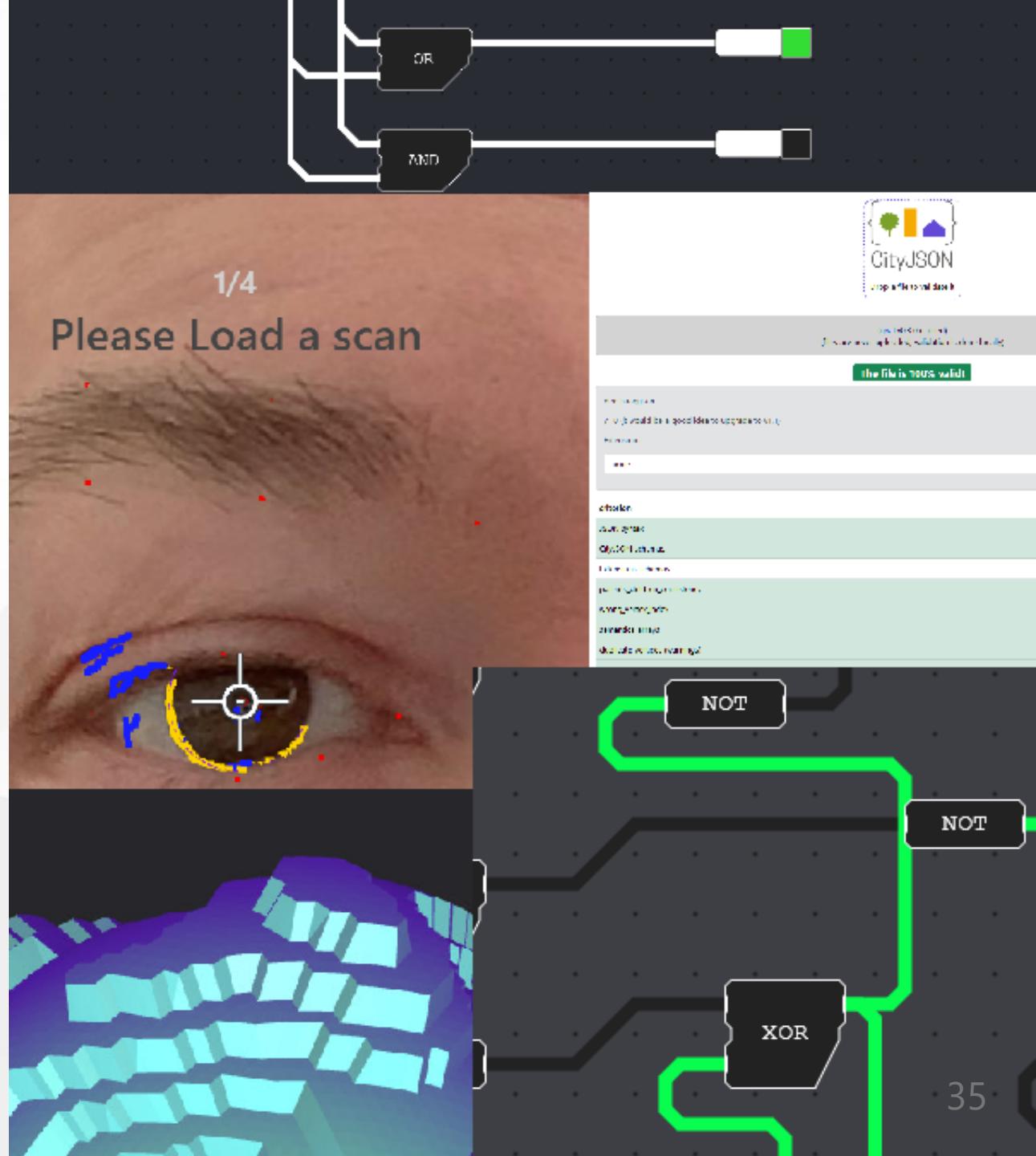
4. Geon Engine

- Personal research project
- Web-based 3D Engine build from first-principles
- 2D and 3D data manipulation



Taken Together

- Insight in geometry processing on the web
- Insight in WebAssembly as a whole
- A good start for the use case application



Lastly, a final word...

Thank you for your attention!

Accessible geoprocessing in the browser using WebAssembly & Visual Programming

by Jos Feenstra