An Interactive Web-based GIS System to Evaluate Hurricane Inundation Impacts

Michael Bednarek [1] **Omid Alemi** [2] **Saeed Moghimi** [3]

[1]Morristown Beard School (High school), 70 Whippany Rd, Morristown, NJ., [2]School of Interactive Arts and Technology, Simon Fraser University, BC, Canada., [3]NOAA National Ocean Service, Coast Survey Development Laboratory, Silver Spring, MD.,

Introduction

The use of web-based geographic information systems (GIS) in a coastal environment can be benecial for both coastal and scientic communities. With the collection of data from numerous sources and social media (e.g. FEMA API's) and with access to model produced results for coastal inundation, we can generalize and interact with models to evaluate the impact of coastal events in an interactive and visual manner.

System design

Bellow the diagram of the complete system from data collection to interaction with the user is given.

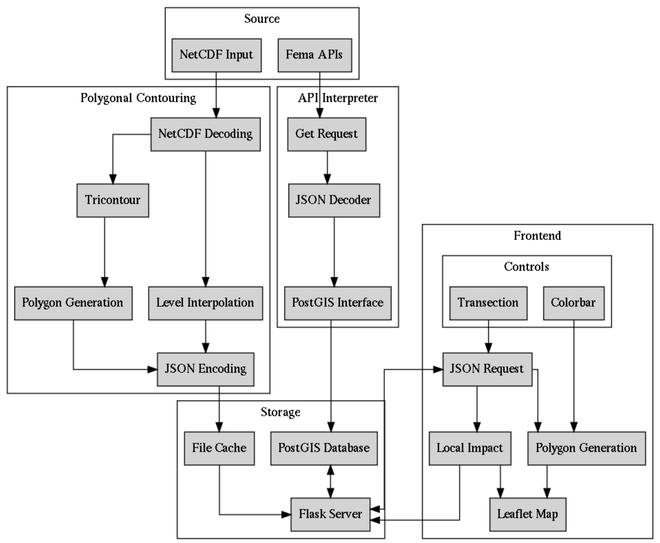

Figure 1: Flowchart of the system.

Data collection

Data from numerous sources including databases from The Federal Emergency Management Agency (FEMA), social media sources such as Instagram, and rst hand collected model produced data from the NOAA's storm surge model results (ESTOFS), are collected to be used in later stages in the system.

Model data processing

- A collection of nodal values for maximum water surface elevation read from model generated NetCDF le.
- For both space saving and ease of manipulation, these values are grouped and vectorized to a collection of polygons.
- The polygons were organized for a given water surface levels.
- These polygons are then encoded into a JSON le including some general information about it's size, value, and resolution.

```
info
    - levels: int                       // number of levels
    - polygons: int                     // number of total polygons
    - vertices: int                     // number ot total vertices
    - min: float                       // minimum of all values
    - max: float                       // maximum of all values

levels: array                           // levels containing values and polygons
    - levels[X]                         // any level
        - value: float                 // value of current level
        - polyCount: int                // number of polygons in the current level
        - polygons: array               // polygon array
            - polygons[X]               // any polygon
                - vertCount: int        // number of vertices in the the current polygon
                - minV: [float, float]  // maximum x and y values for any vertex
                - maxV: [float, float]  // minimum x and y values for any vertex
                - vertices: array       // vertex array
                    - vertices[X]       // any vertex
                        - lat: float    // latitude as float
                        - lng: float    // longitude as float
```

Figure 2: Water surface levels polygons and their associate variables.

Backend

- A PostGIS server accomodates all the data generated from model results and observations collected from data sources.
- A Flask web server is used to serve the compressed information created from the data processing phase. The server provides interface to number of methods for interacting with the collected data.
- The Shaply Python package is utilized to provide general definitions of polygons and other geometrical objects for interacting and generating transects and defining an area of interest to analys land-use in comparison with maximum surge information.

Frontend

The graphical interface is written in pure HTML / JavaScript using Leaflet package to provide an intuitive way of interacting with the data. It allows the user to foremost graph inundation and ood levels, as well as transect values, select regions, and compare with land use information.

Outlook

We are planning to further develop:

- A more advanced polygonal area selection
- A flexible colorbar UI elements
- A direct NetCDF caching
- Implement a seamless connection to data APIs (e.g. FEMA, NOAA)

Github Organization
https://github.com/earth-sys-ai