

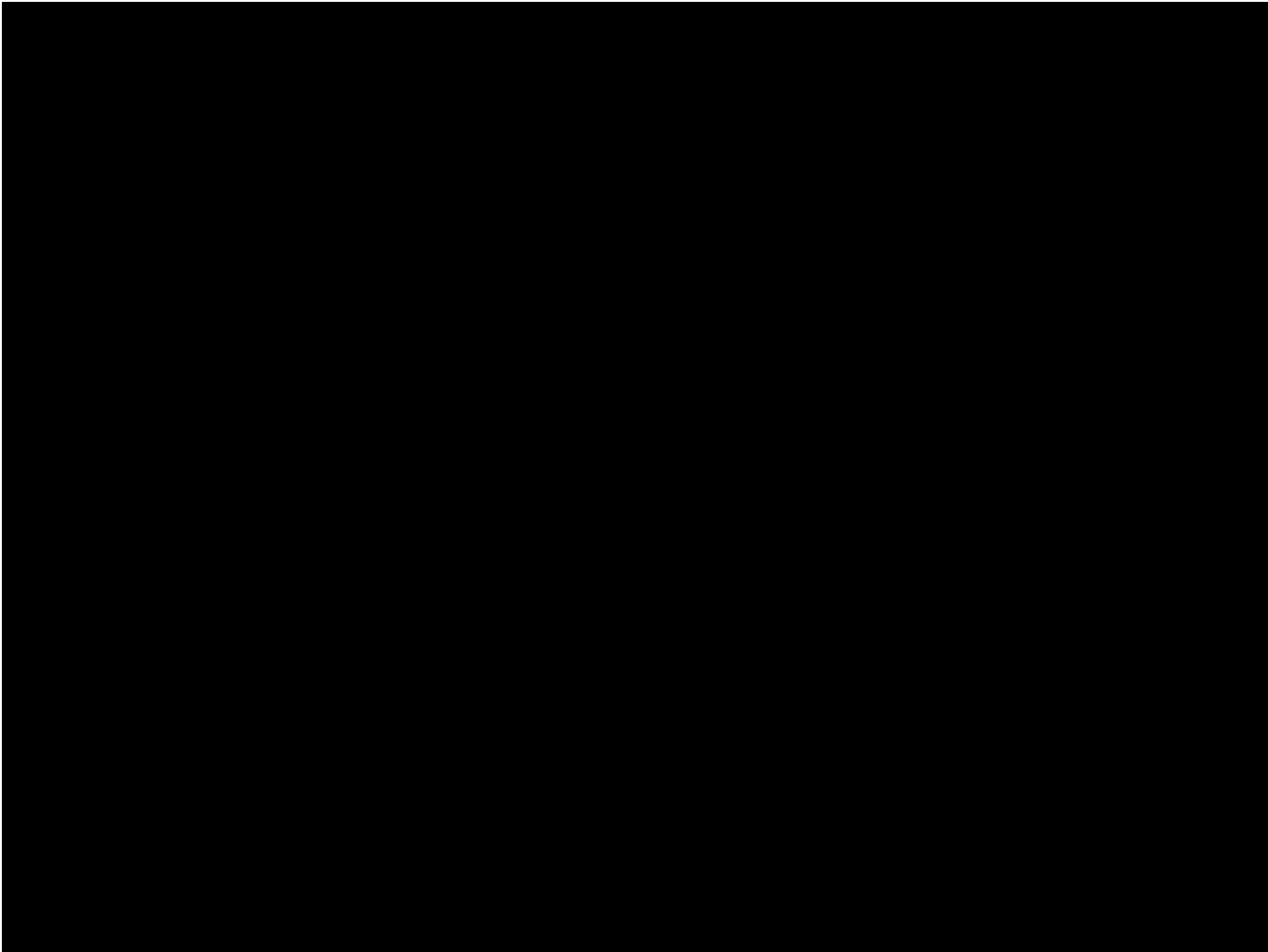
Procedural generation and interactive web visualization of natural environments

Benny Onrust

Rafael Bidarra

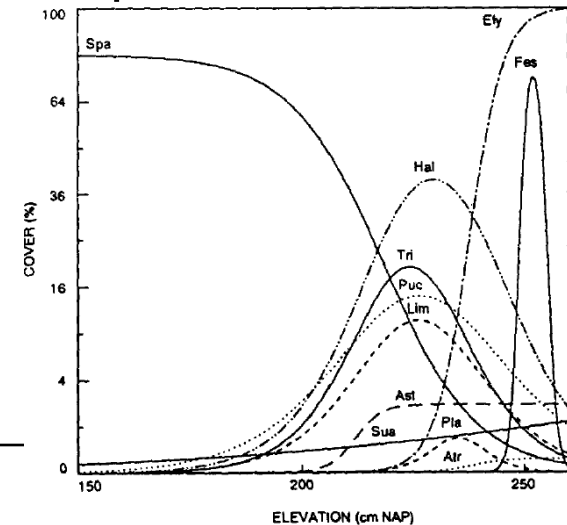
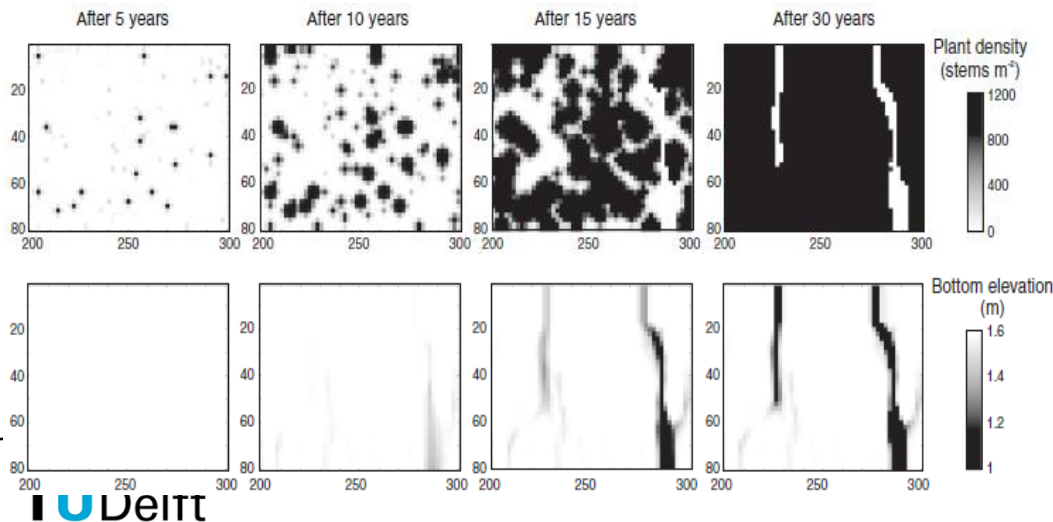
Robert Rooseboom

Johan van de Koppel



Motivation

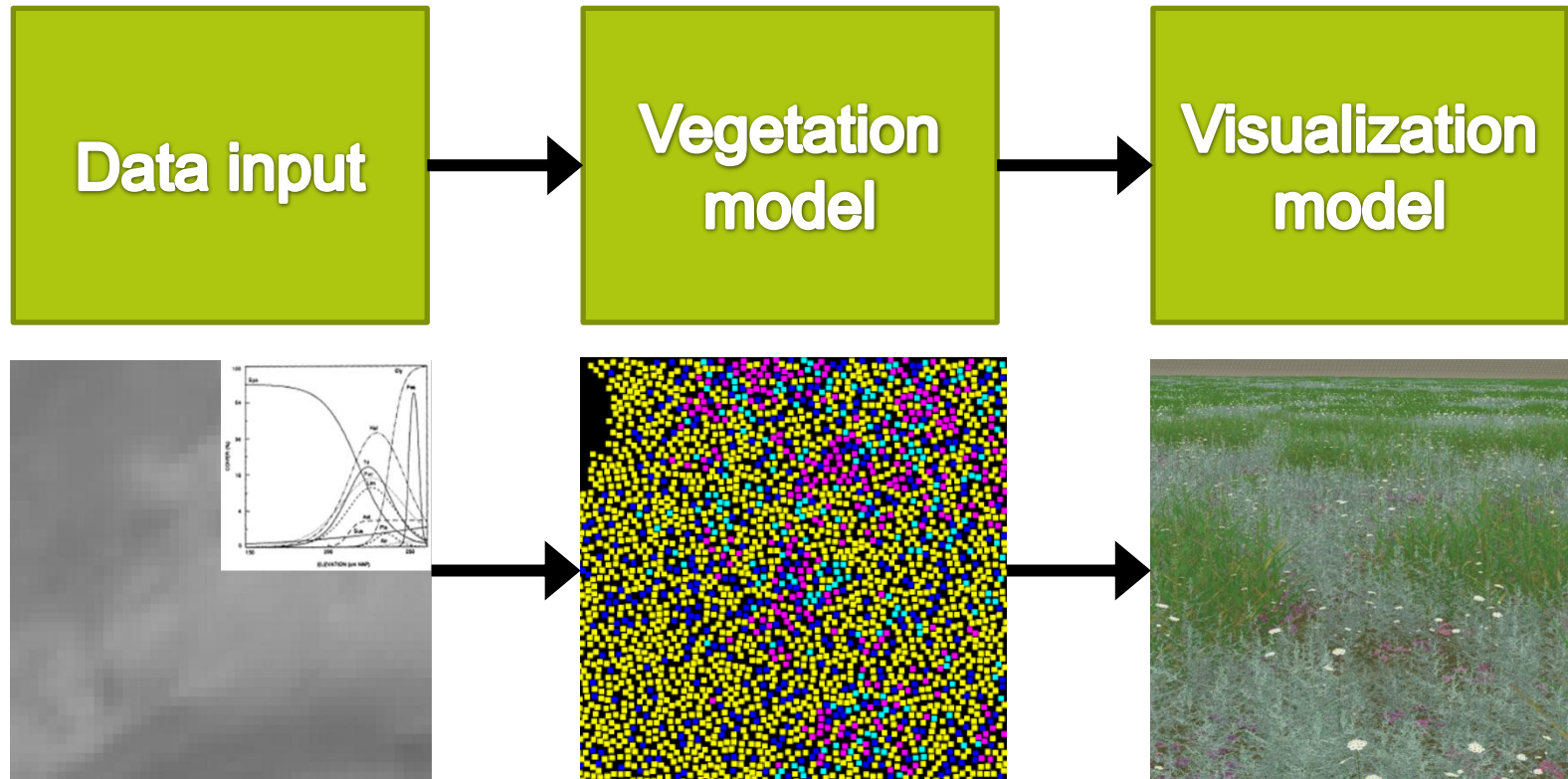
- Ecologists have a lot of different data about natural environments
 - Landscape maps
 - Coverage of plant species
 - Patchiness of plant species
- They want to use 3D visualization to communicate ecological data over the web interactively to the general public



Main questions

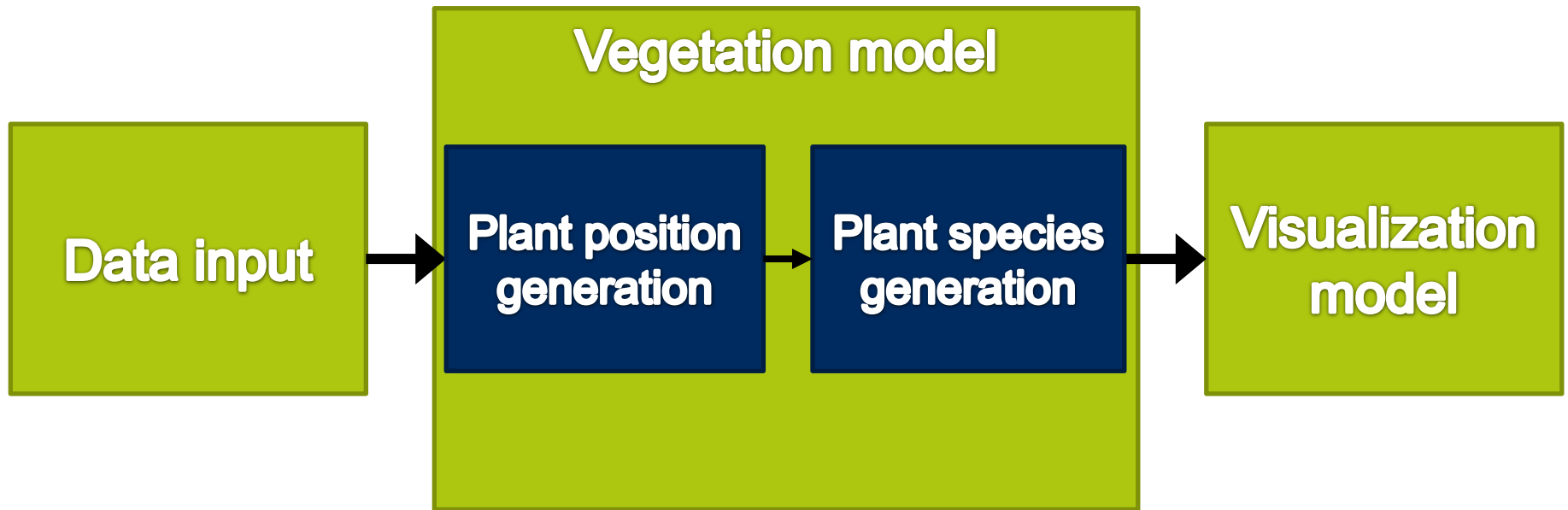
- How to create convincing and correct plant distributions from ecological data?
- How to generate an interactive 3D web visualization of large-scale environments with a large variety of plants?

Approach



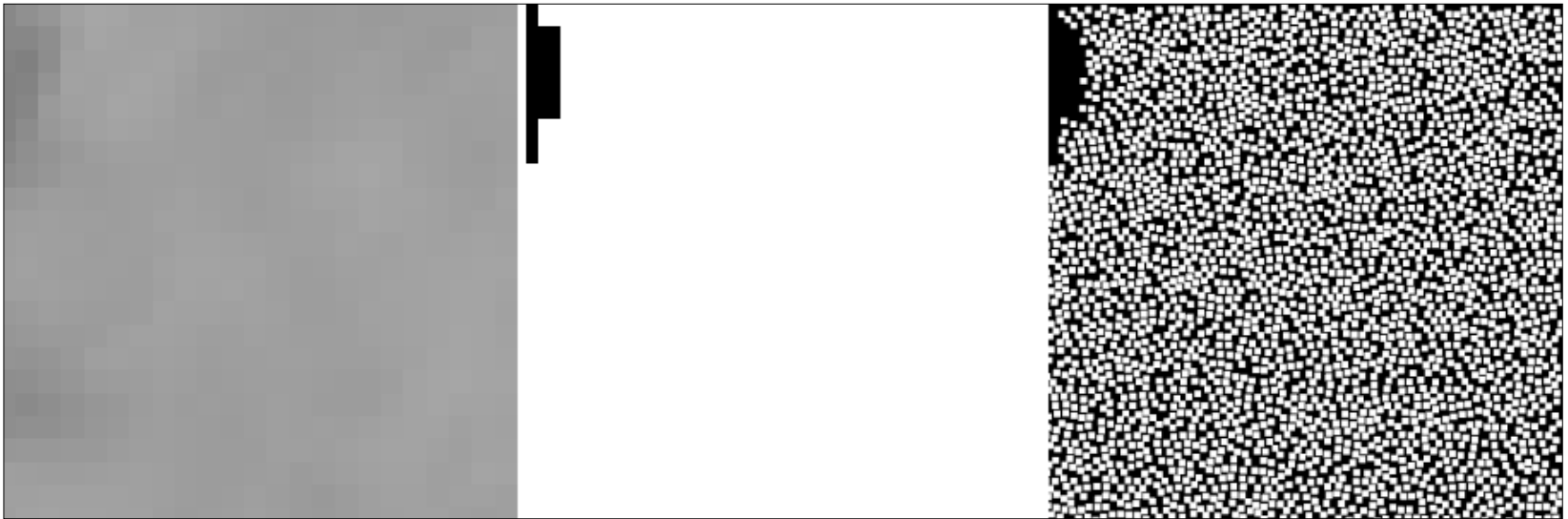
Vegetation Model

- The aim of the vegetation model is to generate a plant distribution from ecological data



Plant position generation

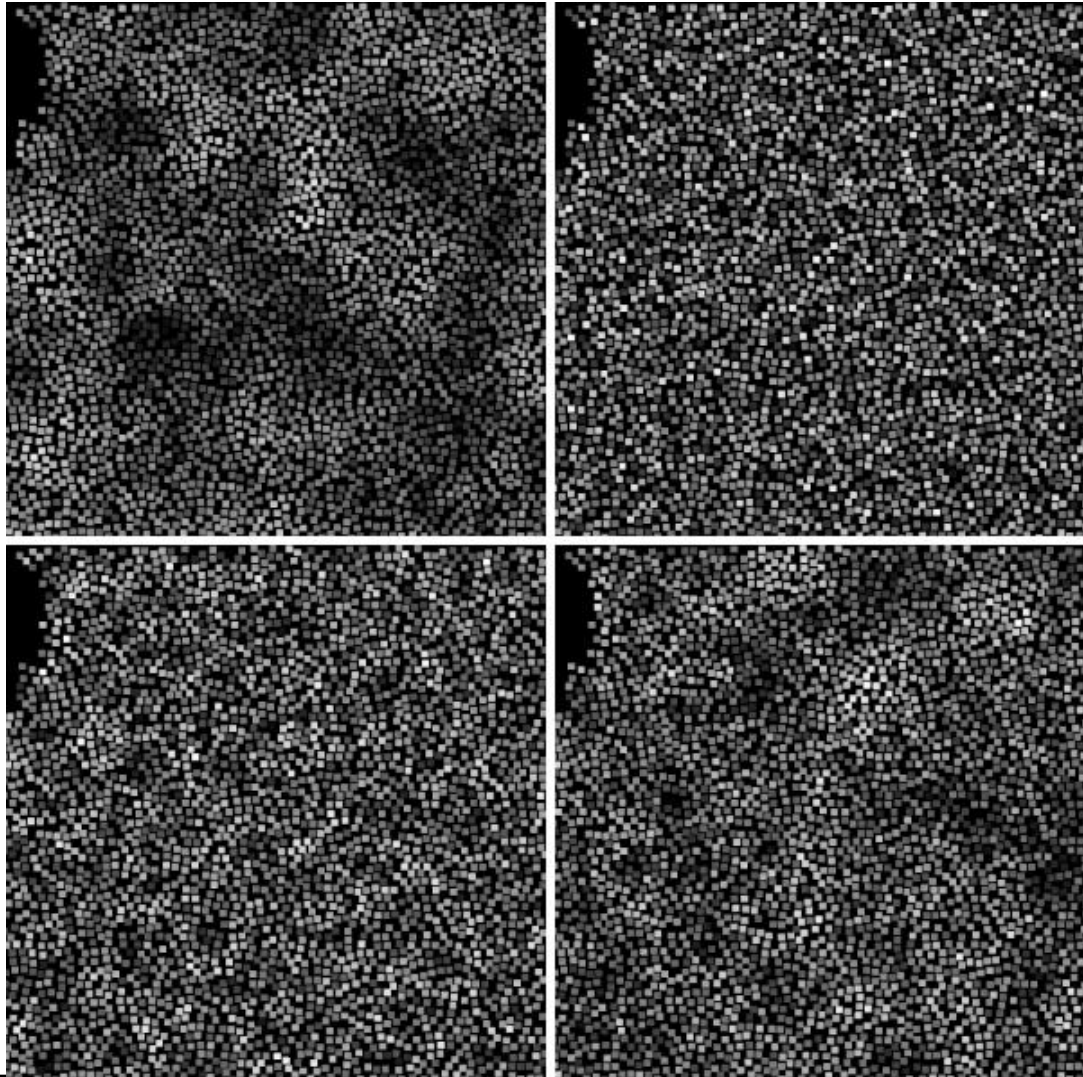
- Aim is to generate all possible plant positions
 - No plant species are assigned during this phase



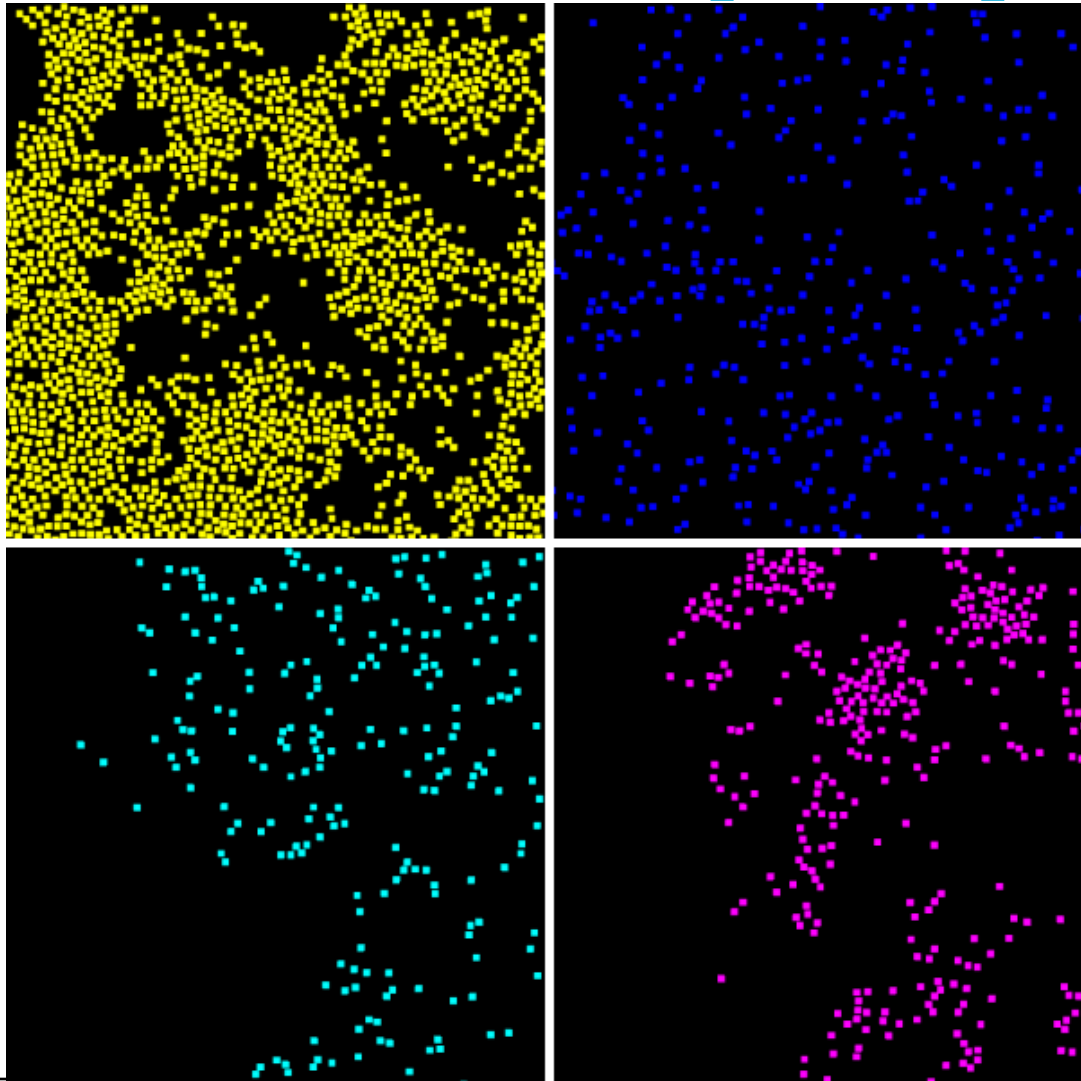
Plant species generation

- Aim is to assign plant species to the generated plant positions using the ecological data about coverage and patchiness
- General process
 - Connect ecological data to each point
 - Generate patterns using fractal algorithm
 - Classification based on coverage and fractal values

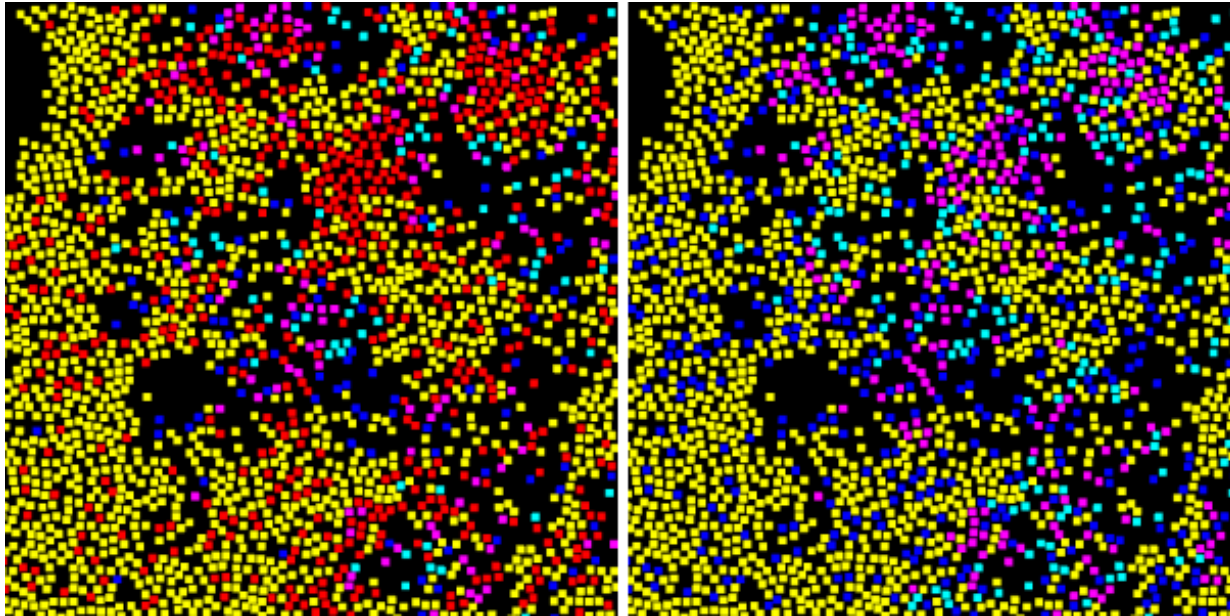
Fractal maps for each plant species



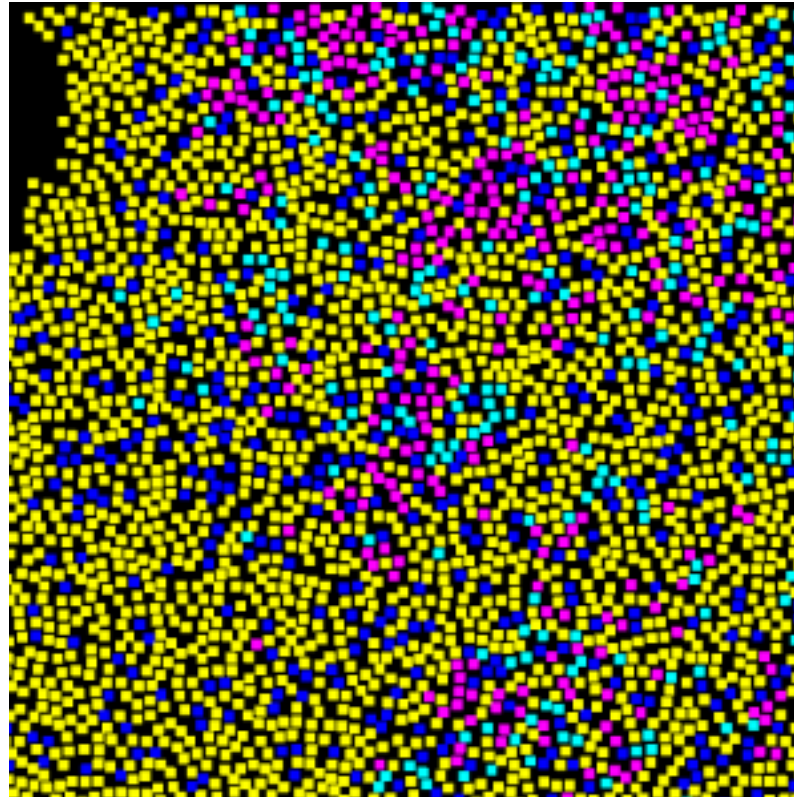
Classification for each plant species



Solve conflicting plant species

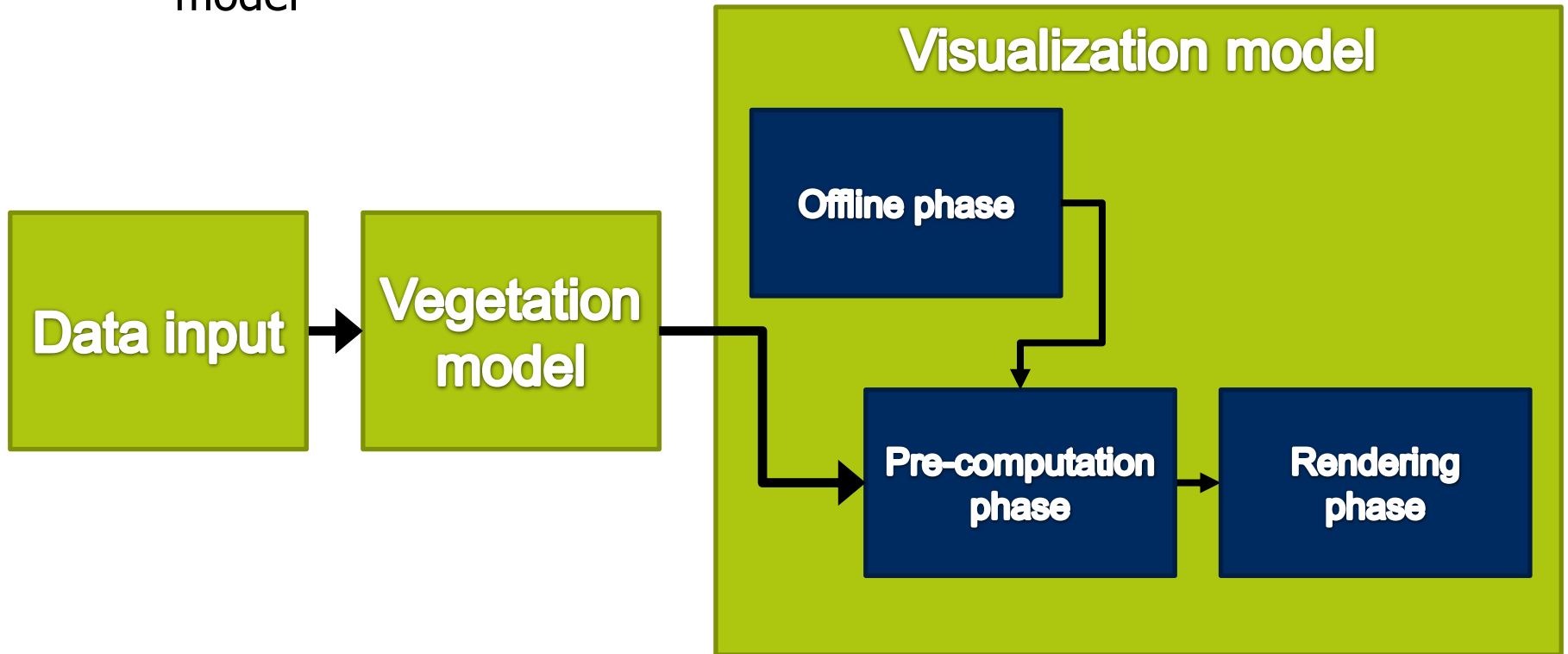


Generated plant distribution



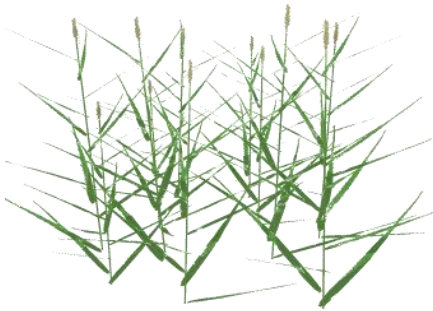
Visualization model

- A WebGL renderer to 3D visualize the result of the vegetation model

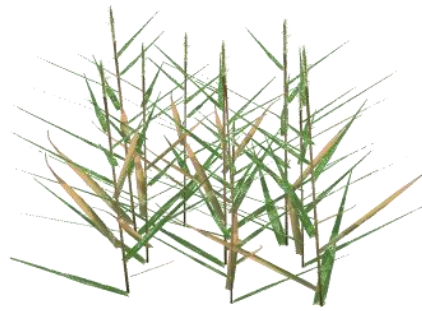


Offline phase

- Plant model generation → L-systems



Elymus Athericus



Spartina Anglica



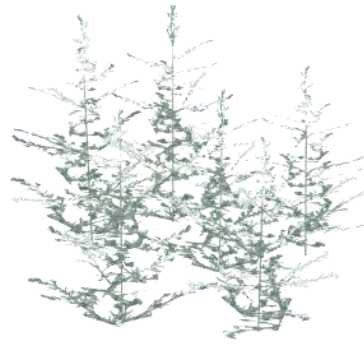
Aster Tripolium



Limonium Vulgare



Atriplex Portulacoides



Artemisia Maritima

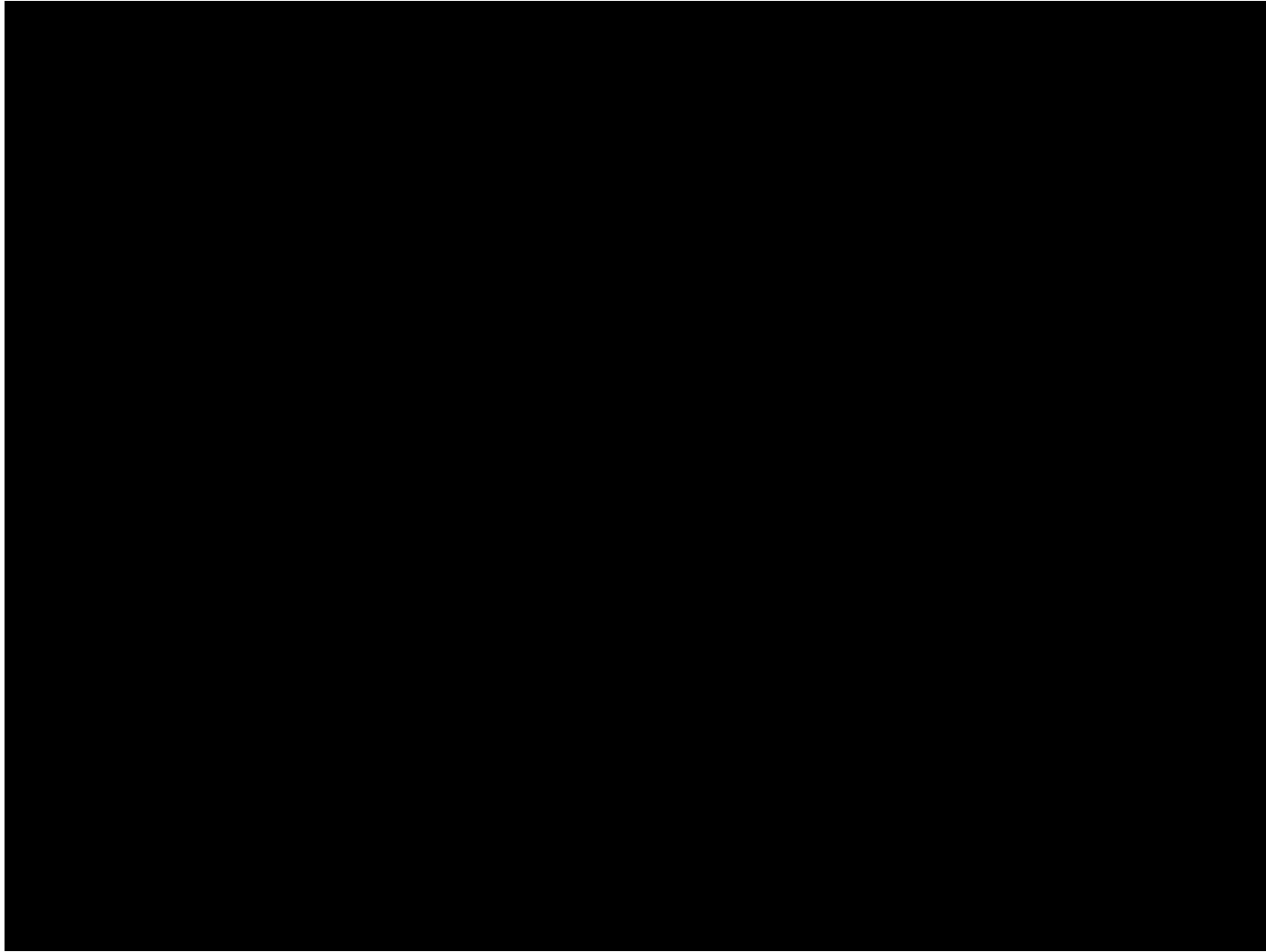


Salicornia Europaea

Pre-computation phase

- Reduce geometry complexity: Levels of Detail
 - Plant models
 - Billboards
 - Terrain color map
- Terrain mesh generation → Height map
- Data is divided in blocks and stored in a quad tree structure
 - Plant distribution
 - Terrain

Rendering phase



Conclusions

- We developed a procedural algorithm that is able to generate convincing plant distribution based on ecological data
- An open-source WebGL renderer that is able to 3D visualize large natural environments with a large variety of plant species
- In the future, we could further integrate the vegetation and visualization model into a single automated pipeline.
- Video, live demo, and source code can be found at <https://graphics.tudelft.nl/benny-onrust/>