SENSEI-Panama
Visualizing animal movement data on a virtual island in cave2

Jillian Aurisano and James Hwang
June 29, 2016

Context

Sensei (SENSor Environment Imaging) is an EVL NSF grant to capture omnidirectional, stereoscopic images and video from real world environments and display this data in virtual reality environments

Sensei-Panama is a project to consider how we might visualize this kind of data in an environment like CAVE2 along with animal movement data to help researchers study animal behavior

Collaborators

Tanya Berger-Wolf at UIC

Meg Crofoot at UC Davis in the Dept of Anthropology and Animal Behavior

They have rich data about Isla Barro Colorado island in Panama and GPS tracking data for kinkajous, spider monkeys, coatis and capuchin monkeys

Research question: understanding how differences in cognition and social structure influence foraging behavior

Big picture motivation

- Old school ecology
  - Ecological field work used to be collected manually
  - Researcher observes behavior of one individual at a time and records what they are doing over a timespan
  - Researcher does a visual sweep of an area and records what everyone is doing
  - But: only can collect a small sample of the complete population, even a small population. Limited by what one person can collect.
- Big data ecology
  - Automated data collection methods from sensors and GPS tracking
  - Can potentially ‘see more’
  - But we have lost the ‘human in the loop’
    - Scientist on the ground
    - Seeing, observing, hypothesizing

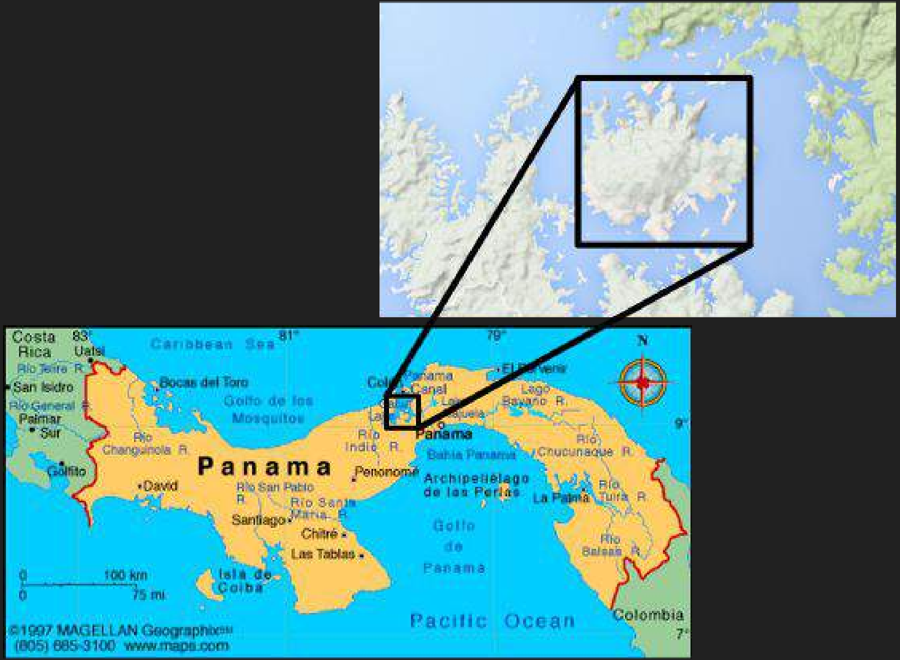
Background

Isla Barro Colorado Island in Panama

Smithsonian Tropical Research Center

“One of the most studied places on earth”

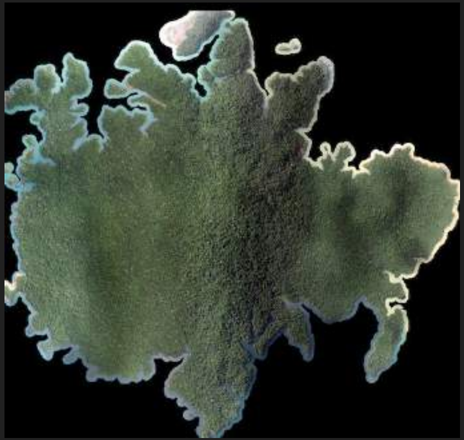
Diverse: Home to 120 mammal species (half of which are bats) 5 species of monkeys, 225 species of ants

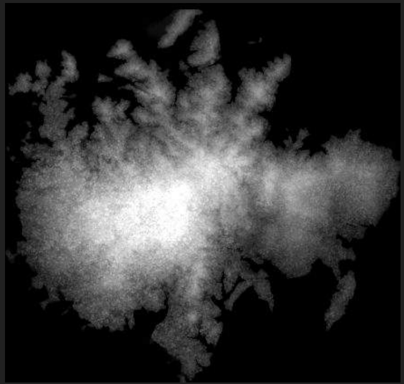


Our data: the Island

Data about the Island

- Current:
  - Aerial Imagery
  - Height map of the canopy
  - Height map of the terrain
  - Identified locations of fruit trees
- Soon ?
  - ? Lidar data that cuts through the canopy to the ground ?
  - Panoramas taken on the ground on the island
- Some day:
  - Sounds, more images, more lidar, paths, structures ...





Our data: Animal movement data from GPS collars

Data about the animal movements

- Latitude, Longitude, Height
  - All with some degree of uncertainty
- 23 individuals
- From capuchin monkeys, spider monkeys, kinkajous and coatis
- Over 30-90 days, every 4 minutes
- Accelerometer data
  - Eventually: used to predict what the individuals are doing (eating, sleeping, fighting)

Analysis tasks:

Questions Meg wants to answer with this data

- Are animals using memory or their senses to find food?
- Do animals follow each other to food sources?
- Do the movement patterns of one species influence the movement patterns of another?

These questions stand to benefit from an immersive, high-resolution visualization approach in an environment like CAVE2

- “Be the animal” and see what the animal can see
- “Follow the animal”, see where they go and what they can see while they move
- Be a researcher on the ground looking at groups of animals and documenting what groups are doing

Visualization Goals

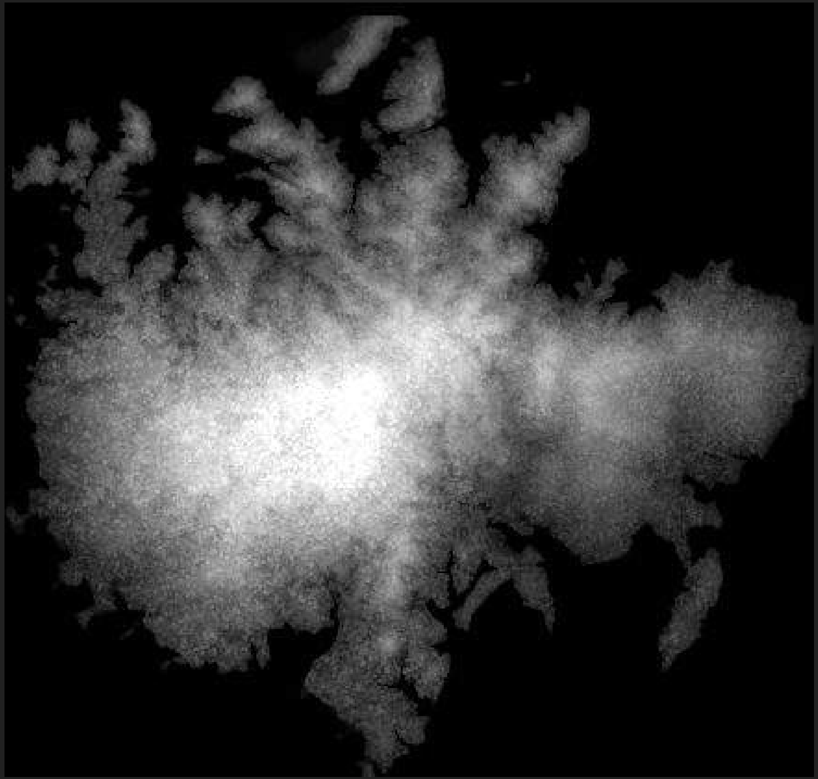
- See how far can we go toward recreating this island from the data in CAVE2?

- What kinds of encodings and interactions for the movement data will help researchers explore the data in an immersive view?
    - Interactive selection and filtering to see one individual at a time, or groups, over different time spans
    - How can we show the movement clearly?
    - What navigation techniques will help?
    - How do we show uncertainty, imprecision and errors from data collection?

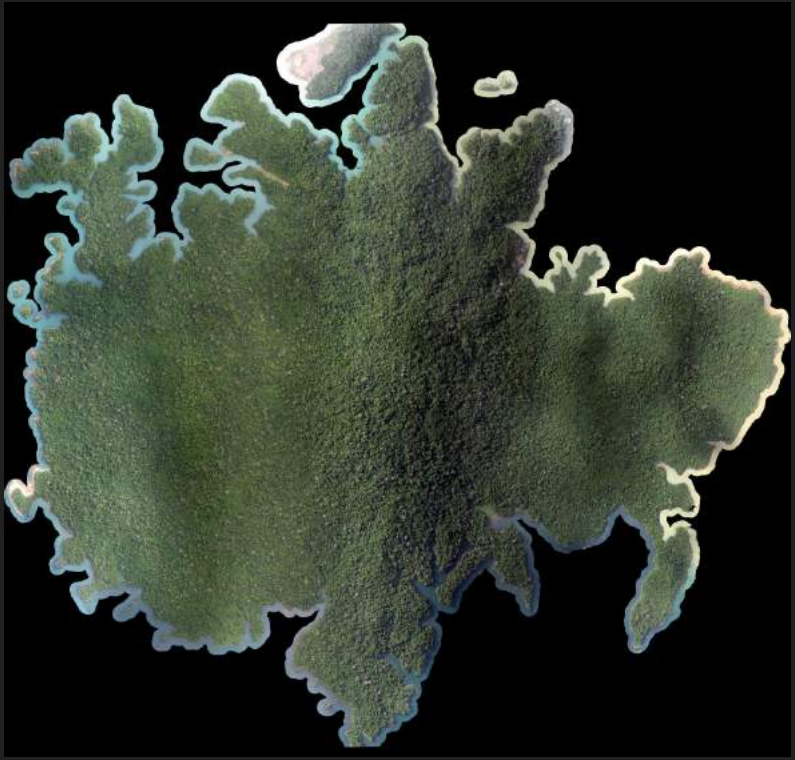
Program Creation Overview

- Addressing the analysis tasks with our data
  - Analysis goal: How can we show whether primates use visual stimulus or semantic memory to find food sources?
  - Creating the island:
    - Examine tradeoffs between high resolution, functionality, and speed.
    - 3D Model
      - Control over 3D Model
  - Visualization of Movement
    - Encoding the movement data
    - Controls
  - Program Feature Overview

From height map of canopy to a point cloud



+



From height map of canopy to a point cloud

Original image: 18cm per pixel

Depth map: 1m per pixel

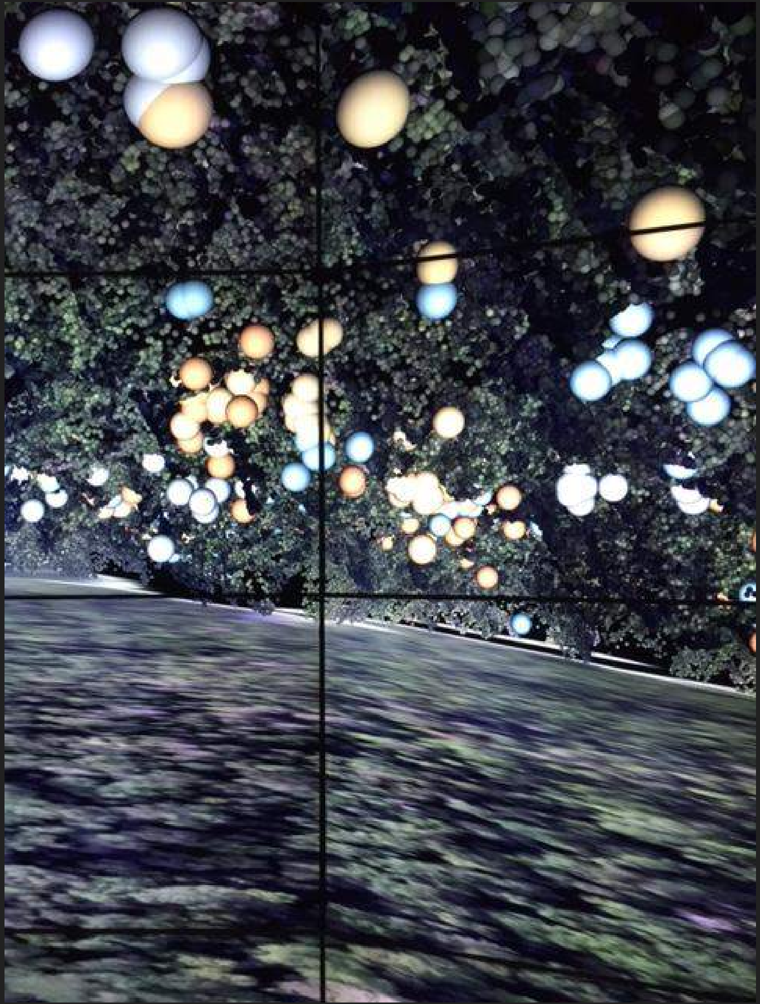
Output point cloud: 36cm per point

File size: over 2.8 GB

Point cloud uses shaders for high performance

Future: seeing if we can get up to full resolution

Point cloud plus ground



Building the terrain from lidar data

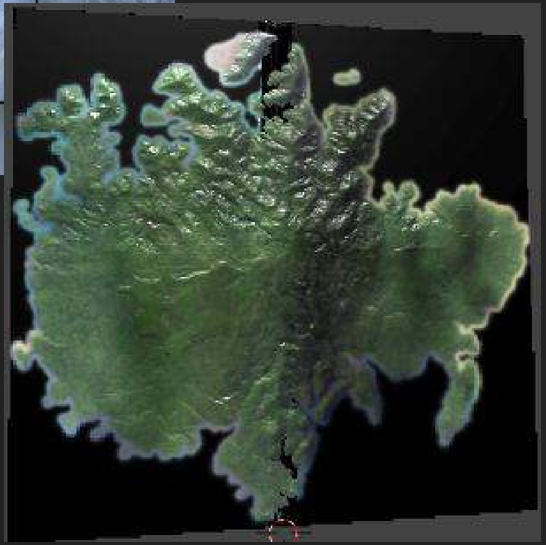
Input: lidar data for the ground

Generating a mesh showing the real heights of the ground

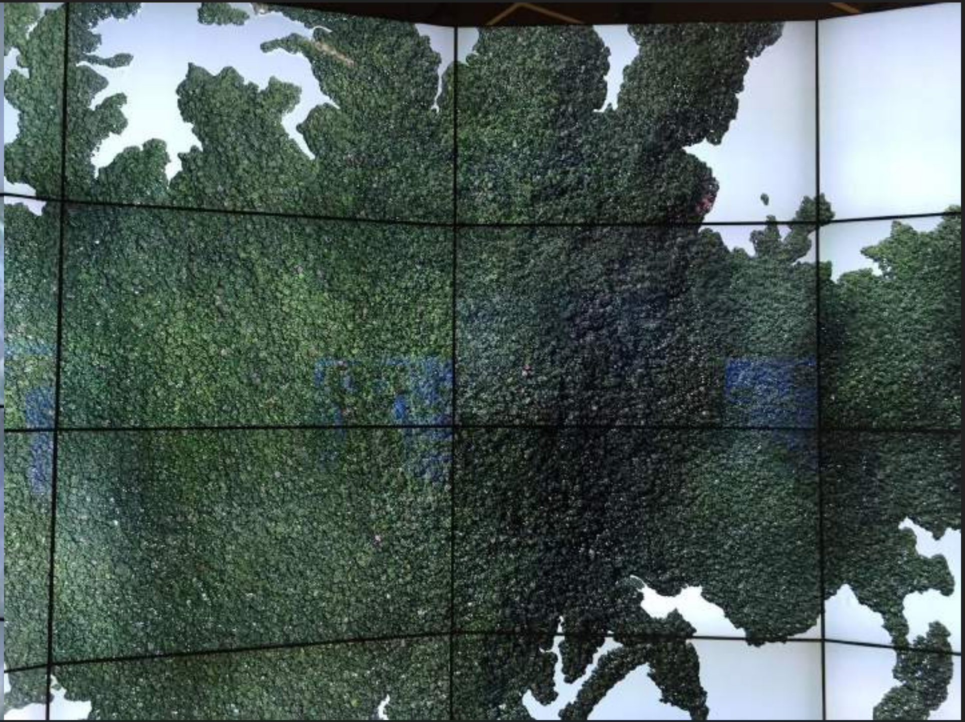
Future: coloring the mesh

- With the Aerial image
- Colored by height?
- Colored by vegetation?

How many faces in mesh do we need to create a good ‘ground-level’ view without limiting performance



Terrain plus point cloud



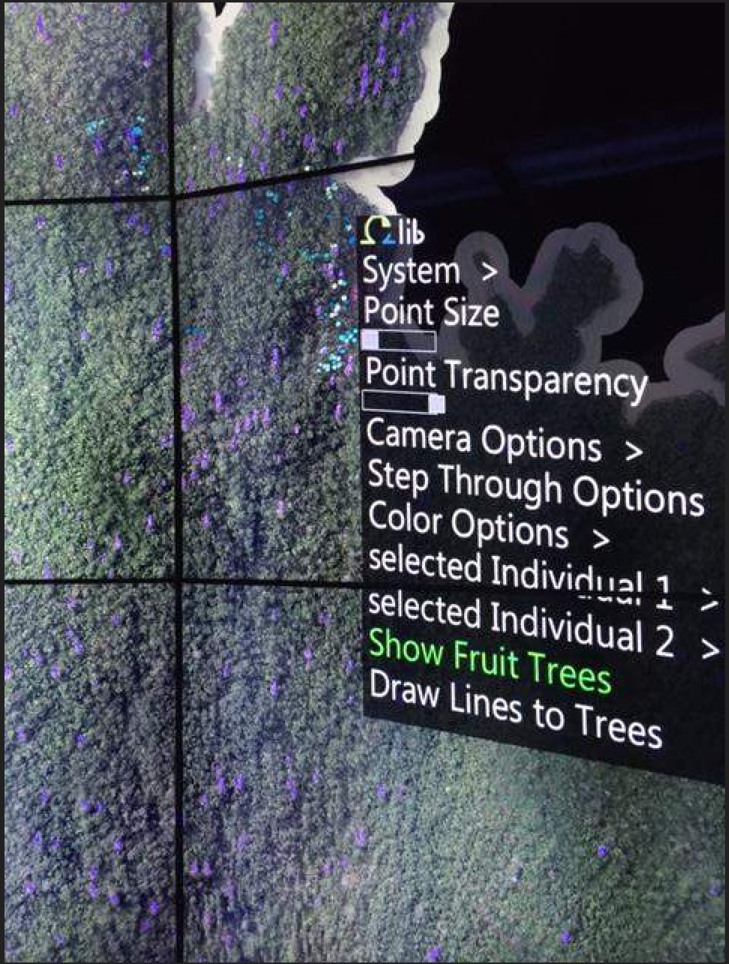
Terrain plus Point Cloud under Canopy

Showing the fruit trees

Why: Meg is interested in how these animals find the fruit-bearing trees on the island

Data file that used the image to find the trees

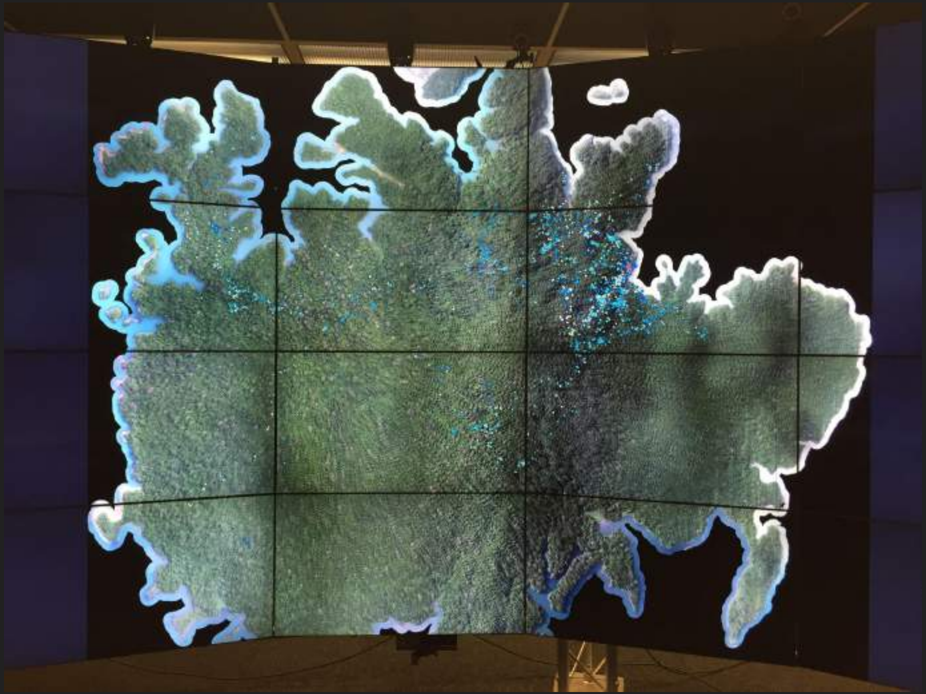
In our code: option to show the trees

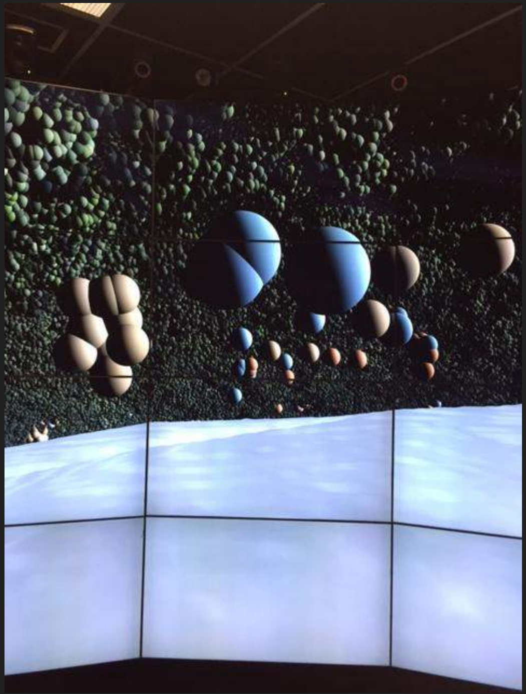


Extra Island View Options

- When researchers is navigating using the wand over the island, can we highlight the nearby fruit trees
- Help researchers see whether fruit trees are in view from a particular point
- How:
    - Currently working on drawing lines from a point (eg. the camera, a monkey) the tops of fruit trees in a radius as you navigate

Visualizing the movement data





Movement data details

- Original movement data was 3GB, preprocessed
- Currently using point cloud and spheres to show each GPS signal
- At the moment just one individual at a time or everyone. Future select several.
- Can select different time spans
    - One day
    - Seven days
    - Whole timeframe
- Color options:
    - Color by time of day, day in a seven day window, day in the total time period
    - Color by individual or species
- We are doing these changes using the shaders
    - Very fast to make changes
- But visually hard to see movement from just the points

Movement data difficulties

- What about other shapes in Omegalib (CPU)
  - Cylinders
    - No easy way to orient the shape.
    - How to quickly filter and color?
  - Line Set
    - Can’t create color gradient. Will be really hard for smoothing.
- Instead of drawing spheres on GPU draw lines
  - Input to the geometry shader changes from points to a line strip.
  - Not working yet
  - Omegalib did not support it until now.
  - Debugging is difficult for Basic OpenGL in Omegalib.
    - Nondescriptive warning message.
    - Code? Omegalib?
  - Solution: Set up and test in an OpenGL environment. If our code works, then it is Omegalib

Future: Navigation

- We want it to be easy to navigate the island
- Modifying the joystick on the wand for improved movement on a terrain
  - Overriding the event handler.
    - Doesn’t work as one would expect.
    - Very odd problems.
- Camera follows the monkey
  - Loop through data to follow the monkey’s movement pattern
- Point to a spot and jump there

Thanks!