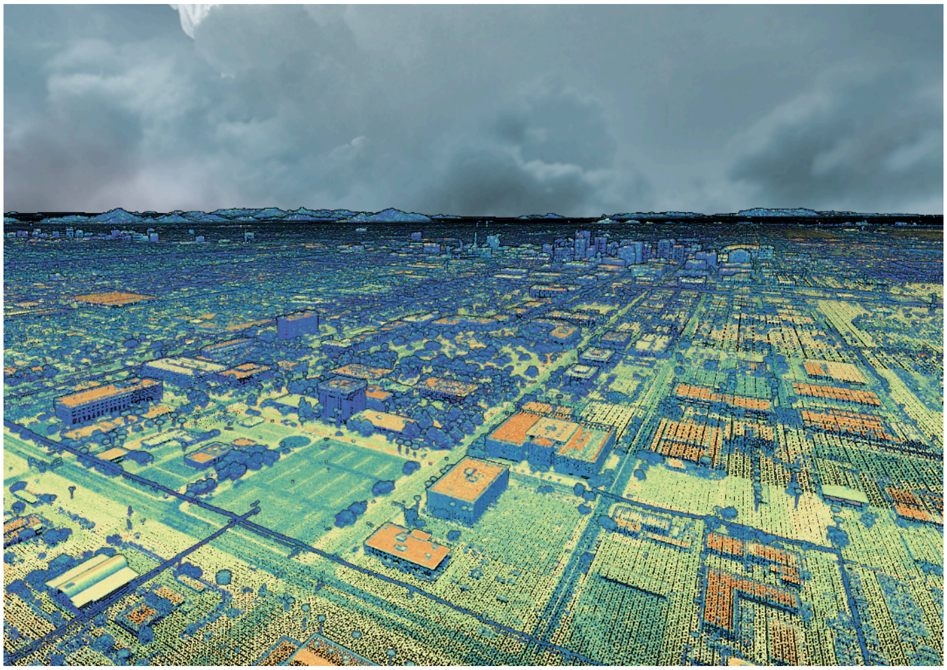
Cameron Dougal and Mary Wright
February 3, 2023
City of Phoenix Office of Heat Mitigation and Response
MIT PKG Center Social Impact Internship

Tree Canopy Detection Using LIDAR



source: USGS / potree.org

Project Purpose

To understand the return on investment of past and future tree planting across the city and identify areas where trees have been removed by computing tree canopy coverage at a high spatial resolution. By designing a processing flow that is reproducible with data sets from past and future years, the change in tree canopy coverage can be compared effectively.

Data Sources Used

- United States Geological Survey (USGS) – 3D Elevation Program (3DEP) LIDAR data – information, access
    - "AZ MaricopaPinal 1 2020" project used in development
- United States Department of Agriculture (USDA) – National Agriculture Imagery Program (NAIP) – Normalized Difference Vegetation Index (NDVI) - access
    - "NAIP_NDVIaggregate_median_2019_pt6mscale" file used in development

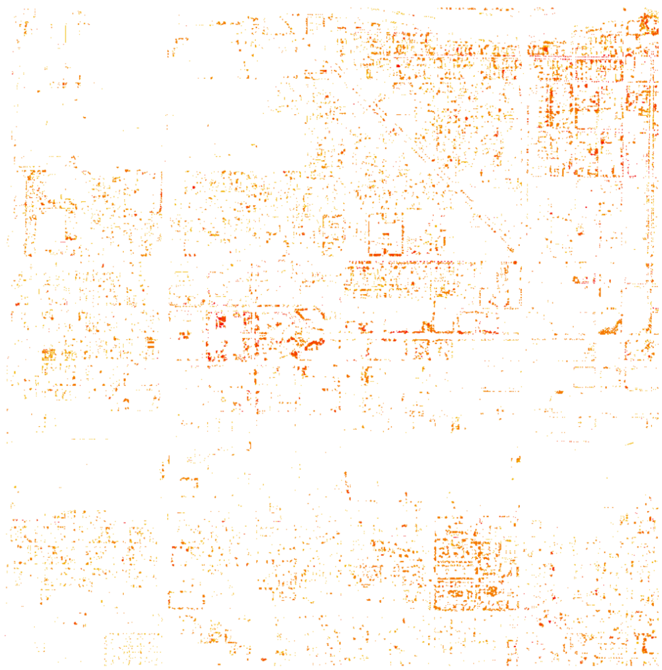
The lidR Package

The lidR package for R was the main driver for the LIDAR data analysis done in the code. Documentation online is sparce, but this resource authored by the package's developers was invaluable for getting going.

The main functions used were to read LAZ files, plot point clouds, compute digital canopy models, manage a collection of LAZ files, and identify/segment individual trees.

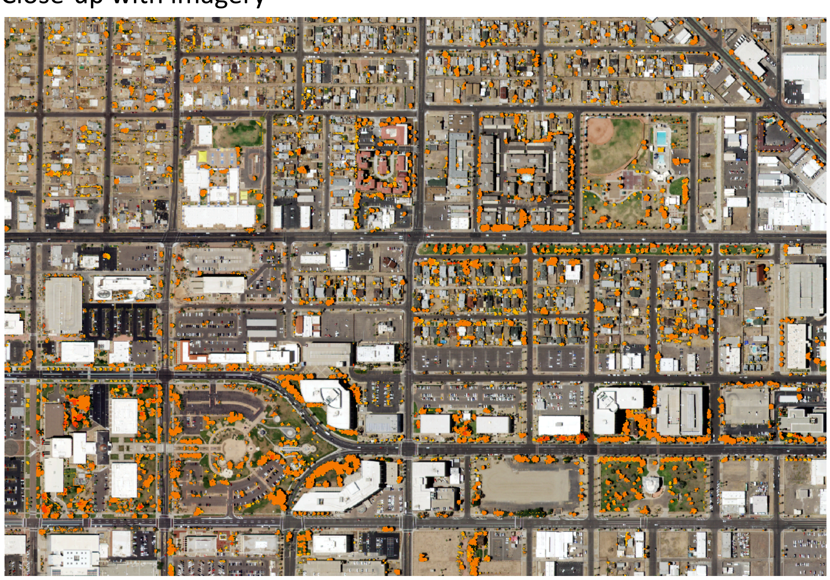
Example Results

The following is an example output raster Canopy Height Model (CHM) representing tree canopy coverage normalized for elevation. The input was nine LIDAR tiles, a 3x3 square surrounding the State Capitol.



Color	Upper value (m)
	≤ 0.01
	≤ 5.0
	≤ 15.0
	≤ 25.0
	≤ 35.0

Close-up with imagery



The CHM can then be further analyzed to extract locations of tree tops and/or segmented crowns.

Outline of Process

See the comments in the code for a detailed breakdown of each step in the workflow.

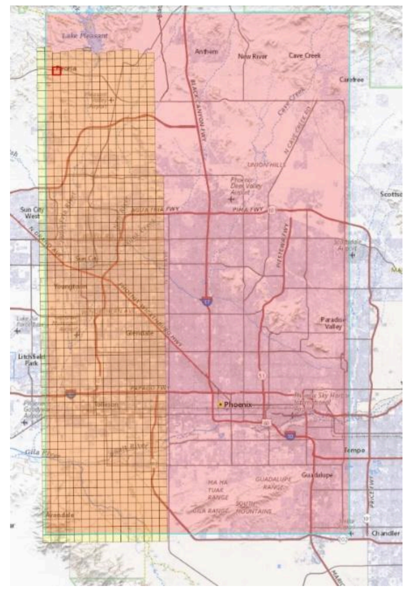
Future Steps

Batch Downloading Tiles

The code contains a small section of code for downloading the LAZ files in bulk. Though it does successfully download the files, it corrupts them in some way that prevents them from being used by the lidR package. For the example results above, the tiles were downloaded manually using the viewer.

Scaling Up

Although the code works on small test regions, scaling the process up to the area of the entire city will require significant file storage and processing time.



Each LIDAR tile is about 100-150 MB. The city is about 63x45 tiles = 2835 tiles * 116 MB = 354 GB. At typical download speeds, this can mean 5-10 hours of download time. The CHM raster output files are much smaller at about 870 KB per tile or 2.46 GB overall.

The total processing time for each tile is about 2 minutes, meaning around 95 hours of processing time on an average computer for the entire batch of tiles. For this reason, it may only be feasible to do small batches of tiles. Otherwise, the code may need to be further optimized for efficiency, or processing time on a more powerful computer should be rented.

The USGS LIDAR data is also hosted on the cloud, opening the possibility of cloud computing aka “streaming” the data instead of downloading it and processing locally (see information 1, 2).

Segmented Canopy Analysis

The output raster from the tree segmentation function (dalponte2016()) can be further analyzed to find the average crown width or average crown area. This could be done with a raster processing package such as terra.