APPENDIX A:
Algorithms for Debris Volume and Water Depth Computation

Access Restoration Project Task 1.2 Report 2 (of 2)
Algorithms for Debris Volume and Water Depth Computation
Prepared by:
Digital Imaging and Remote Sensing Laboratory
Chester F. Carlson Center for Imaging Science

1. Overview

This report details the algorithms developed for automatically calculating the volume of roadway debris piles and the depth of water in the roadway.

The report is broken into five sections:

- Introduction: a brief discussion about the motivation for calculating debris volume and water depth
- Methods: mathematical descriptions and examples of what each step of the algorithm is accomplishing
- Results: how the algorithms performed on test data sets and the level of accuracy that can be expected in future use
- Conclusions: a summary of the report, algorithms, and results on test data
- Deliverables: a description of the deliverables for Task 1.2

2. Introduction

In the Access Restoration Project Task 1.2 Report 1, the algorithms for detecting roadway debris piles and flooded areas were described in detail. Those algorithms take CRS data as input and automatically detect the roadway obstructions. Although the location of obstructions are a valuable information product, it is beneficial to take the processing a step further and characterize the obstructions in terms of size.
For debris piles in the roadway, this means calculating the volume. For flooded portions of the roadway, this means calculating the depth of the water. Not only does this additional information allow someone to look at a map of the obstructions and determine their severity, but it also feeds into the Access Restoration module to determine restoration times and optimal routes.

3. Methods

The debris volume and flood depth estimation algorithms will be described separately in their own sections.

3.1 Debris Volume Estimation

Debris detection results in a set of debris pile point clusters – the lidar points that sample the true surface of the debris pile. The volumes of these debris piles are then estimated one by one. First, the DEM points that fall within the boundaries of the 2-D projection of a debris pile are appended to the pile for improved volume estimation. This is because it is

assumed that each debris pile sits flush on the terrain below it. Each pile is then modeled as a 3-D surface using alpha shapes. Alpha shapes are parameterized by an alpha radius, α, that controls how tightly the surface wraps around the set of points. For volume estimation, it is important that the alpha radius is large enough to form a closed surface around the points. Setting α to $3 \cdot \mu_k$ (where μ_k is the mean distance between a point and its k neighbors, as described in Task 1.2 - Report 1) is sufficient for generating closed surfaces around the debris piles. Once a debris pile has been modeled using alpha shapes, it can be treated as a set of tetrahedrons whose individual volumes are computed and summed to find the total volume for the pile. This process is carried out for each debris pile. Figure 1 shows a set of roadway debris piles in Haiti before and after alpha shape reconstruction.

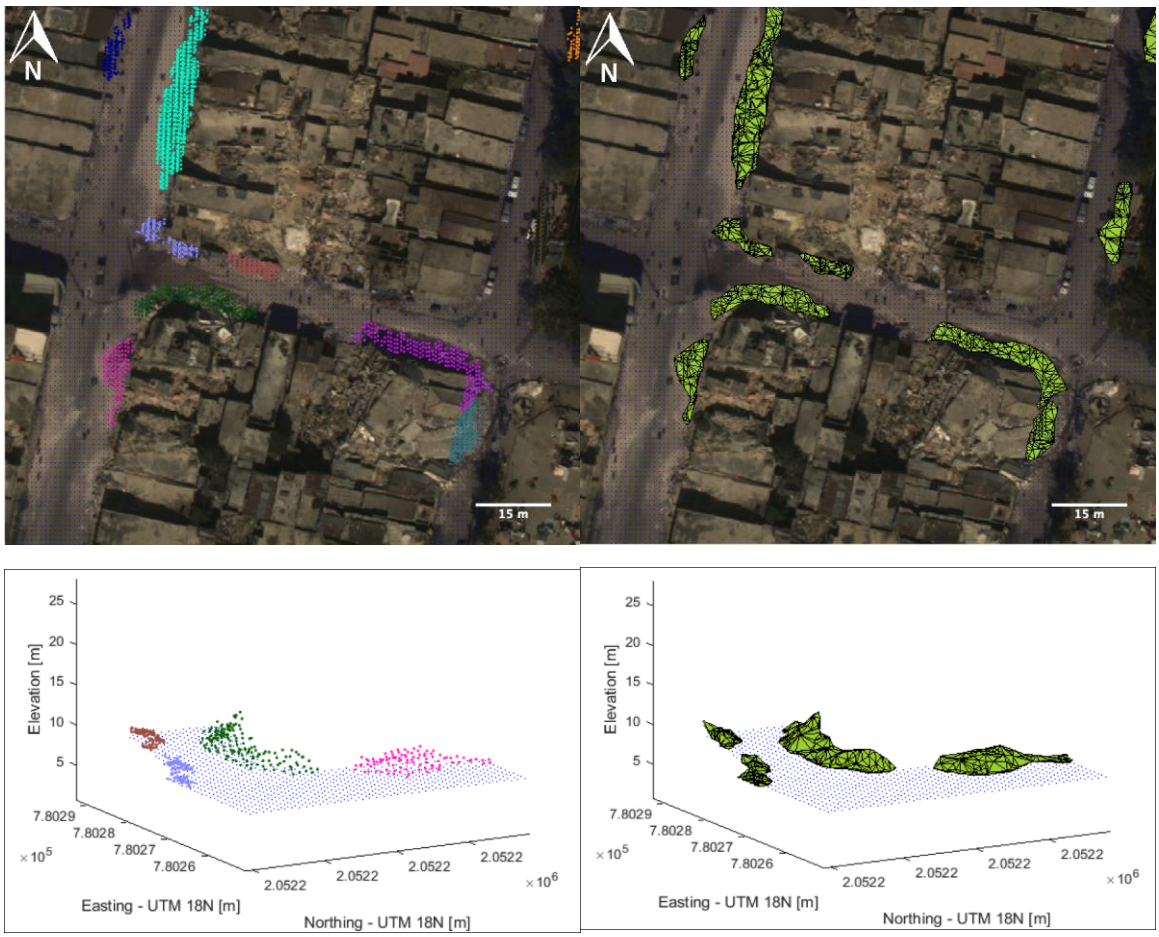


Figure 1: A set of detected roadway debris piles from Haiti are reconstructed as alpha shapes for volume estimation. The images above show an overhead view overlaid on imagery and 3-D views are shown below. The colors in the top left image correspond to those in the bottom left view.

3.2 Flood Depth Estimation

Inputs to the depth estimation stage include the road mask (produced from the road network shapefile in the detection stage), the flood classification image (produced in the detection stage) and a digital elevation model (DEM) raster that covers the same region as the flood image that was input in the detection stage (it can cover a larger region as long as it fully covers the flood image). The DEM image is converted to the same coordinate system as the flood image and then resampled to match its resolution. Figure 2 displays a classified flood image and a corresponding resampled DEM.

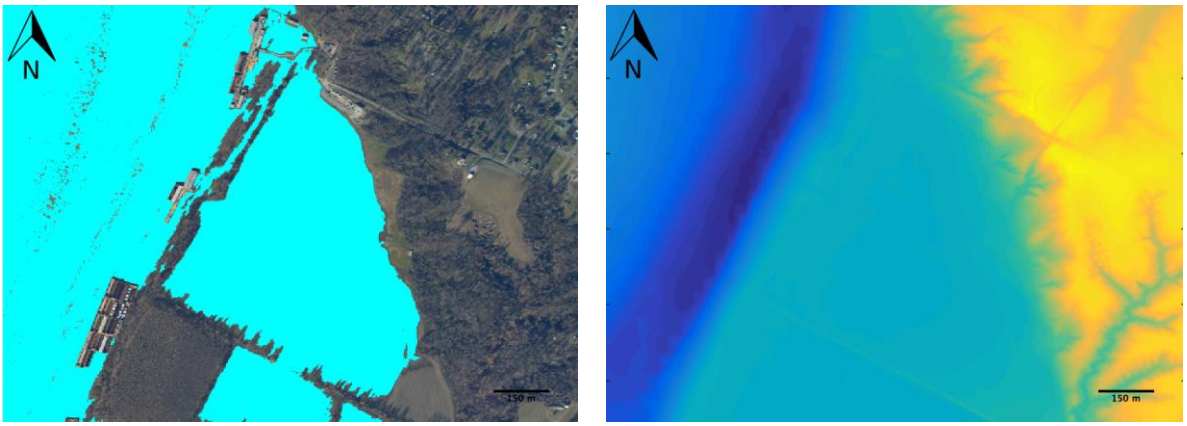


Figure 2: A classified flood image and the corresponding digital elevation model.

Rather than estimate the depth for all flood pixels at once, the depth is estimated for each connected component separately. This follows the assumption that continuous regions of water sit at the same level. The flood image is converted into a label image in which connected component is assigned a unique integer. Figure 3 shows the label image for the flood map in Figure 2.

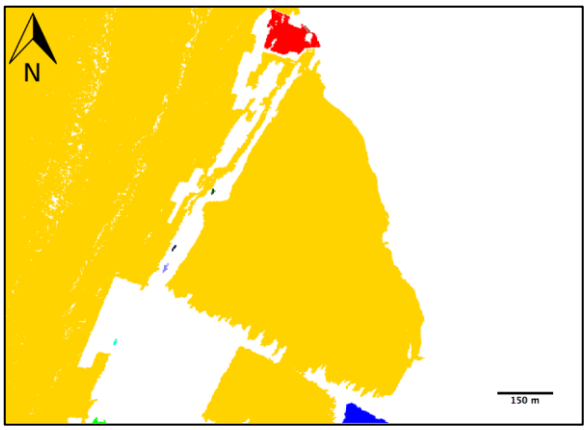


Figure 3: Each connected component of the flood pixels is assigned a unique integer (shown as different colors).

The next step is to find the water depth for each flood pixel in the roadways. This is accomplished by finding the elevation at the boundary between water and land for each connected component. We are only interested in connected components that touch the roads, so the other components are discarded. There are several hundred or thousand pixels

at the border of each connected component, but we only want to take the elevation from one. The pixel with the max elevation is chosen because it is typically the most accurate measure of the water level. The difference in elevation between each DEM pixel and the pixel at the water border is equivalent to the water depth, so all of the DEM pixels in the connected component are subtracted from the max elevation pixel. Figure 4 shows the max elevation pixel selected for an image as well as the max elevation point on a 3-D view of the DEM. The 3-D view illustrates that the calculated max elevation point is indeed on a raised ridge, confirming that it is a high point at which the water meets the land.

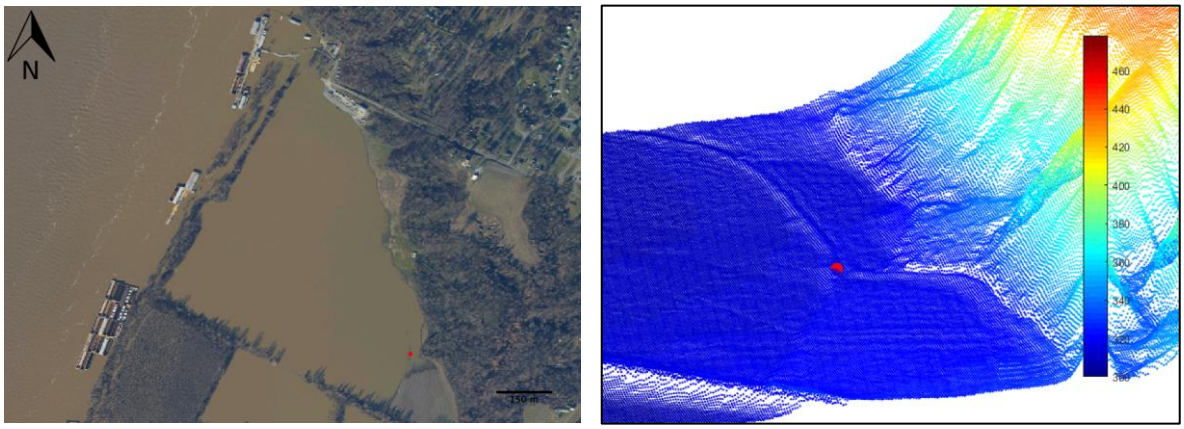


Figure 4: The maximum elevation at which the water meets the land is calculated and represented as a red circle in both an overhead image and the DEM point cloud. From the 3-D view it can be seen that the point lies on a raised ridge. The scalebar on the DEM corresponds to elevation in feet.

Figure 5 shows the calculated water depths (in feet) overlaid on top of the regular image. It should be noted that the water depths are only valid for regions that are land flooded by water. Regions that are already water (in this case, the Mississippi River) have an elevation value of zero in the DEM, and therefore blow up to really large values when subtracted from the max elevation. This could be avoided by limiting the depth image to only the roadways, but that would also throw out a lot of valid flood depth information, so it is retained instead. When the depths are exported to shapefiles (addressed in Task Report 1.4), only the roadway water depths are included. Figure 6 shows the actual depth image that the algorithm produces (the overlay is just for visualization here).

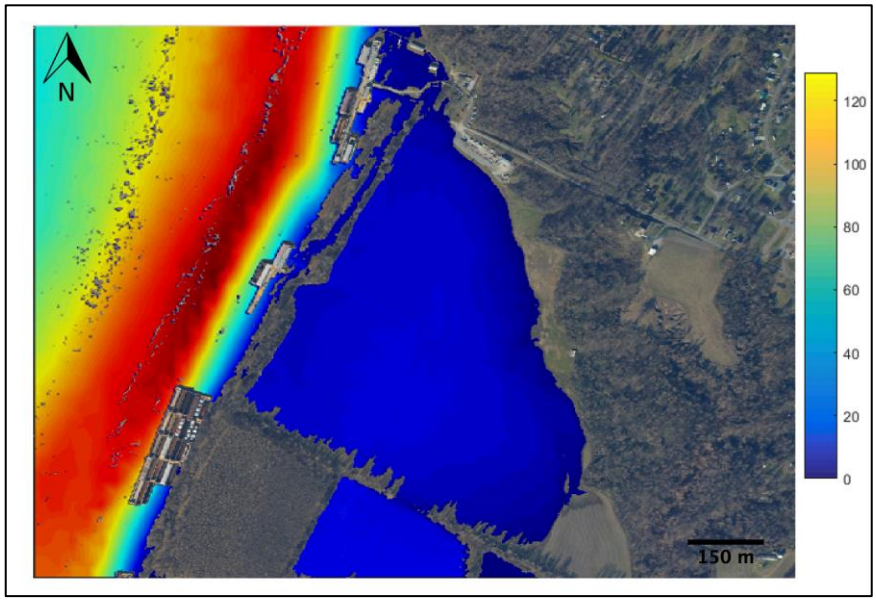


Figure 5: A depth map for regions of the image that were classified as flooded. The colorbar shows the water depths in feet. The depths are only valid for actual land regions (the extremely large depths on the left of the image are caused by DEM values of zero in the Mississippi River).

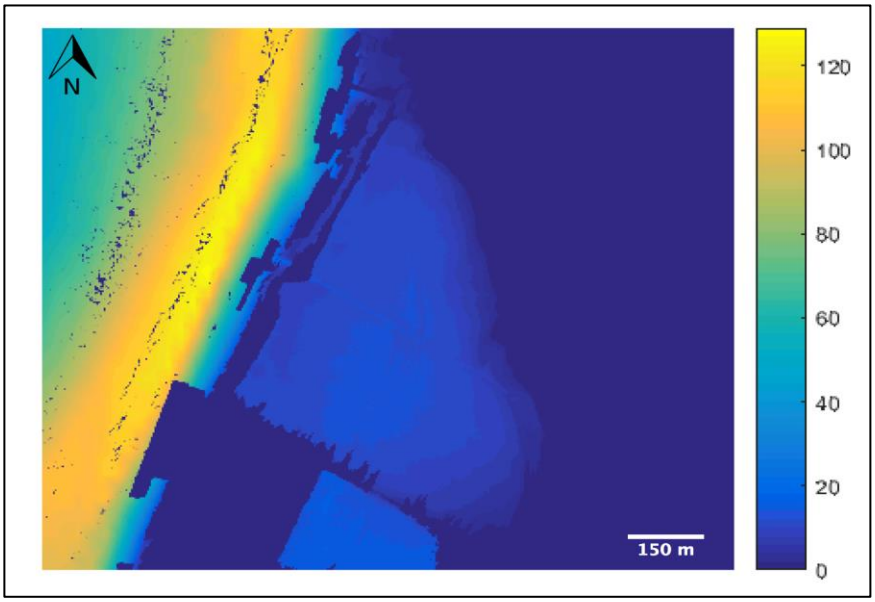


Figure 6: The depth map produced by the algorithm. Pixels that were classified as non-flood have values of zero. The colorbar shows the water depths in feet.

4. Results

4.1 Debris Volume Estimation

There is no direct way to validate the volumes estimated for the debris piles in Haiti because no field data were ever published for the volume of individual piles. Even if individual pile volume data were published, it would have to have been measured within a few days of the lidar data were collected to be consistent. Rather than validating the volumes directly, an experiment was conducted to measure the agreement between volumes obtained from high resolution, ground-based scans and airborne scans of debris piles from construction sites. This method characterizes the volume estimates from airborne lidar data in comparison to what can be obtained on the ground. Terrestrial and airborne laser scans were collected of debris piles at two locations in Rochester, NY. Figure 7 shows an image and lidar scan of one of the construction sites. Figure 8 shows both airborne and terrestrial point clouds of the same debris pile to highlight the difference in point densities of the two modalities.

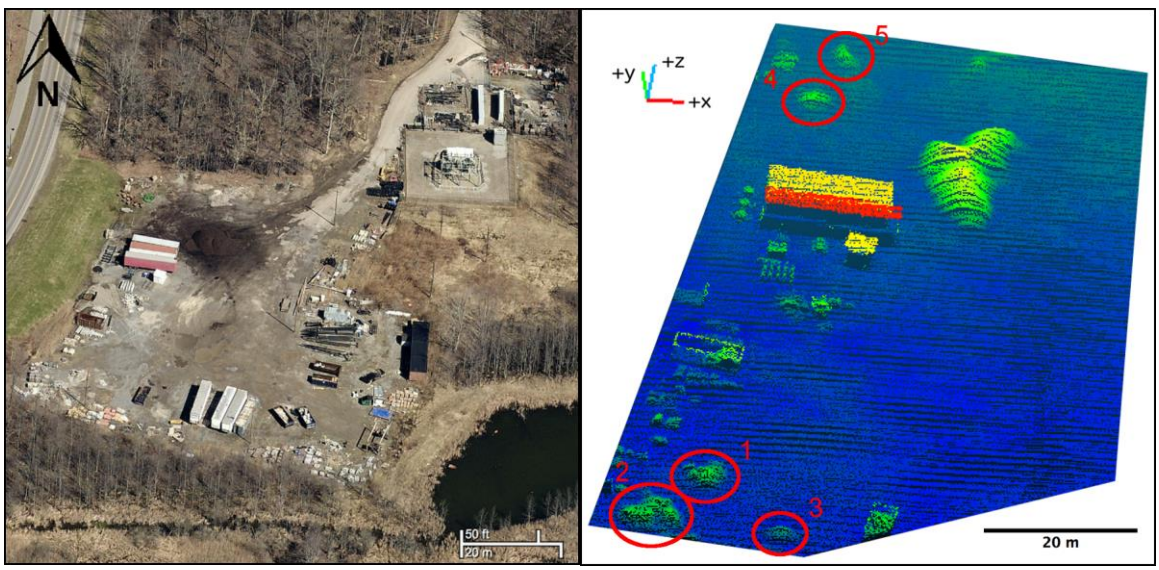


Figure 7: One of the two construction sites in Rochester, NY. The image was taken two months before the lidar data were collected, so the contents of the site differ in many spots. The five debris piles used in this study are circled in red and numbered.

Each of the two airborne point clouds (one for each construction site) was input to the debris detection and volume estimation algorithm. The debris piles are in parking lots, rather than roads, so no road vector input was used in this case. The algorithm produces a set of debris pile point clouds as well as a set of alpha shape reconstructions of the piles. The algorithm detected many more debris piles than were actually scanned (only a portion of the debris piles at each site were scanned due to time constraints), and in some cases, piles that were in close proximity to one another were segmented as one large pile. To be able to accurately compare the volume estimates, only the piles matching the ground-based piles could be

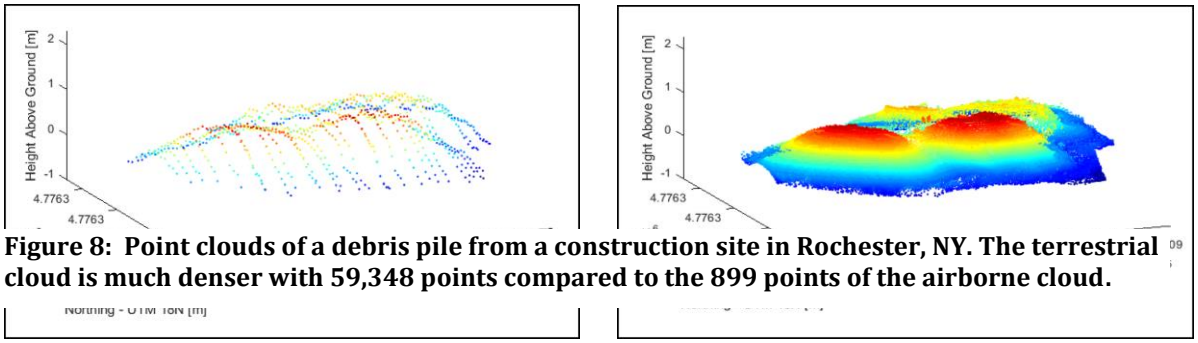


Figure 8: Point clouds of a debris pile from a construction site in Rochester, NY. The terrestrial cloud is much denser with 59,348 points compared to the 899 points of the airborne cloud.

considered. To ensure that this was the case, the piles were segmented using the bounding polygons of the ground-based piles and then reconstructed again as alpha shapes. Once the piles are reconstructed as alpha shapes, the volume is computed by simply summing the volumes of the individual tetrahedrons that make up the alpha shape.

The debris piles from the terrestrial lidar scans are already segmented into individual piles so they cannot be used as inputs to the algorithm. Instead, they are simply reconstructed as alpha shapes and the volumes are obtained. A comparison of the volumes computed for the airborne and terrestrial scans is presented in Table 1.

At full density (52.3 pts/m^2 and 20.6pts/m^2 for lots 1 and 2, respectively) the algorithm calculated volumes with a mean error of 1.25 $\pm$ 0.9 m^3 with a maximum error of 3.33 m^3. The absolute value of the volume differences was used to obtain the mean error. This translates to a mean error of 9.83 $\pm$ 8.58%. If the analysis is limited to debris piles with a ground-measured volume of 20 m^3 or higher (this is the minimum volume threshold used for the detection testing) the mean volume error becomes 1.71 $\pm$ 1.06 m^3 translating to a mean error of 4.26 $\pm$ 3.47% (see Table 2). Debris-clearing rates of typical machinery can be in excess of 200 m^3/hr so an estimation error of around 3 m^3 would likely be insignificant to a mathematical model used for prioritization of response activities. These results suggest that at these resolutions, airborne lidar scans can be used as inputs to the proposed algorithm and obtain reasonably accurate estimates.

Table 1: A comparison of volume estimates for debris pile point clouds obtained using terrestrial lidar and airborne lidar.

Pile ID	# Points (Ground)	# Points (Air)	Volume (Ground) [m^3]	Volume (Air) [m^3]	Volume Difference [m^3]	Volume Difference [%]
LOT1 PILE1	37367	529	10.26	10.78	-0.52	5.03
LOT1 PILE2	129688	1132	22.62	22.82	-0.20	0.88
LOT1 PILE3	74199	313	4.13	3.41	0.71	17.33

LOT1 PILE4	87192	398	5.09	4.74	0.35	6.84
LOT1 PILE5	43891	303	3.04	2.79	0.25	8.21
LOT2 PILE1	107566	1210	75.81	77.78	-1.97	2.60
LOT2 PILE2	96333	1813	100.19	100.58	-0.39	0.39
LOT2 PILE3	110248	1088	47.23	50.00	-2.77	5.86
LOT2 PILE5	37382	188	5.82	3.95	1.87	32.07
LOT2 PILE6	59348	899	25.44	24.00	1.44	5.68
LOT2 PILE7	53419	346	12.00	11.36	0.65	5.38
LOT2 PILE8	47209	787	32.45	29.12	3.33	10.27
LOT2 PILE9	56451	202	5.22	4.46	0.76	14.59
LOT2 PILE10	60483	814	25.25	26.99	-1.75	6.92
LOT2 PILE11	35678	224	6.27	5.44	0.83	13.22
LOT2 PILE12	182391	2168	123.11	121.24	1.86	1.51
LOT2 PILE13	33535	168	7.06	5.23	1.83	25.89
LOT2 PILE14	17205	210	6.71	5.74	0.96	14.34
				Mean	1.25	9.83
				St. Dev.	0.90	8.58

Table 2: A comparison of volume estimates for debris pile point clouds obtained using terrestrial lidar and airborne lidar. Only debris piles with a ground-measured volume of 20 m^3 are included in this table.

Pile ID	# Points (Ground)	# Points (Air)	Volume (Ground) [m^3]	Volume (Air) [m^3]	Volume Difference [m^3]	Volume Difference [%]
LOT1 PILE2	129688	1132	22.62	22.82	-0.20	0.88

LOT2 PILE1	107566	1210	75.81	77.78	-1.97	2.60
LOT2 PILE2	96333	1813	100.19	100.58	-0.39	0.39
LOT2 PILE3	110248	1088	47.23	50.00	-2.77	5.86
LOT2 PILE6	59348	899	25.44	24.00	1.44	5.68
LOT2 PILE8	47209	787	32.45	29.12	3.33	10.27
LOT2 PILE10	60483	814	25.25	26.99	-1.75	6.92
LOT2 PILE12	182391	2168	123.11	121.24	1.86	1.51
				Mean	1.71	4.26
				St. Dev.	1.06	3.47

Although the results presented in Table 2 suggest that the proposed algorithm is able to produce accurate estimates of debris pile volumes at full resolution (52.3 pts/m^2 and 20.6pts/m^2) it is necessary to investigate the accuracy of estimates at the resolution of the Haiti point clouds (4.2 pts/m^2 on average). To accomplish this, the point clouds were randomly subsampled from 20.6 pts/m^2 to 2 pts/m^2 in increments of 2 pts/m^2 and the same volume comparisons were performed. Only debris piles with a ground-measured volume of 20 m^3 or greater were considered for this analysis. Figure 9 contains plots of the results for mean volume error in terms of both m^3 and percent. At the highest point density (20.6 pts/m^2) errors of around 1.75 m^3 or 5% are obtained, and at the lowest point density (2 pts/m^2) the errors are about 9.25 m^3 or 22%. At the mean point density of the Haiti data (4.2 pts/m^2) the error is about 5.75 m^3 or 14%. These results suggest that while higher density data is certainly beneficial, reasonable results can still be obtained at low point densities. A mean error on the order of $\approx$ 6 m^3 will not have a significant effect on the computation of restoration times.

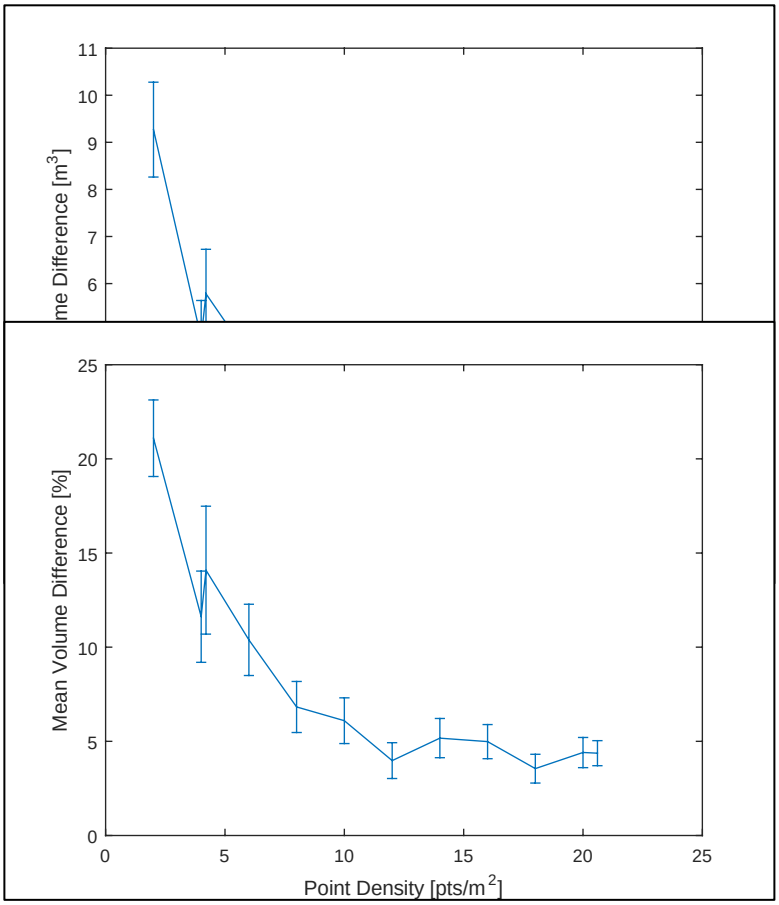


Figure 7: (Top) Mean volume error in m³ as a function of point density. (Bottom) Mean volume error in % as a function of point density. Only debris piles with a ground-measured volume of 20 m³ were considered for this analysis.

4.2 Water Depth Estimation

Unfortunately there is no way to validate the water depths estimated in the roadways of the Mississippi River flood imagery. To our knowledge, no roadway water depth maps were ever published. Even if they were, they would have to match up with the day the imagery was collected. Despite the lack of validation data, there are some logical checks that can be used to verify that the estimates are approximately correct. For example, the water depth should be approximately zero right at the land-water border. If the algorithm is producing very small depths at the border region (i.e., the first few pixels away from land) then it is matching what one would expect. The depths for the rest of the pixels in the same connected component can then be considered as accurate as the elevations from the DEM

raster, because that is what they are based off of. Figure 9 illustrates this concept. The depth *d1* is calculated by simply subtracting the pixel elevation *h1* from the elevation at the land and water border, *h0*. As long as the algorithm correctly finds *h0*, then it is safe to assume the rest of the depths are correct. We can check that *h0* is nearly correct by examining the smallest depths (*d1*, *d2*, etc.). Figure 10 shows an area of a flooded roadway at the very edge of the flood region. A depth profile is plotted for the pixels along the green line. The water depth stays flat at zero until it hits the border and then slowly increases

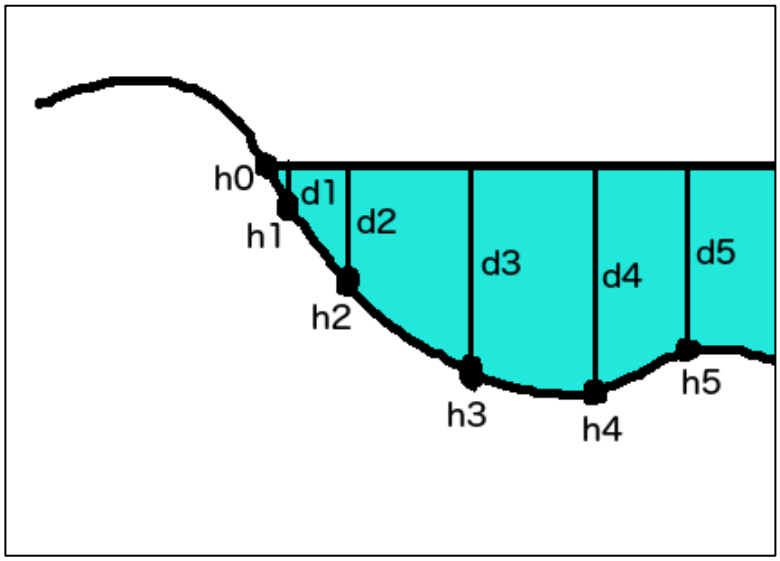


Figure 8: An illustration of water depth estimation. The point h0 is the point where water meets land, d1 is the depth of the water equal to h0-h1, d2 is the depth of the water equal to h0-h2, etc.

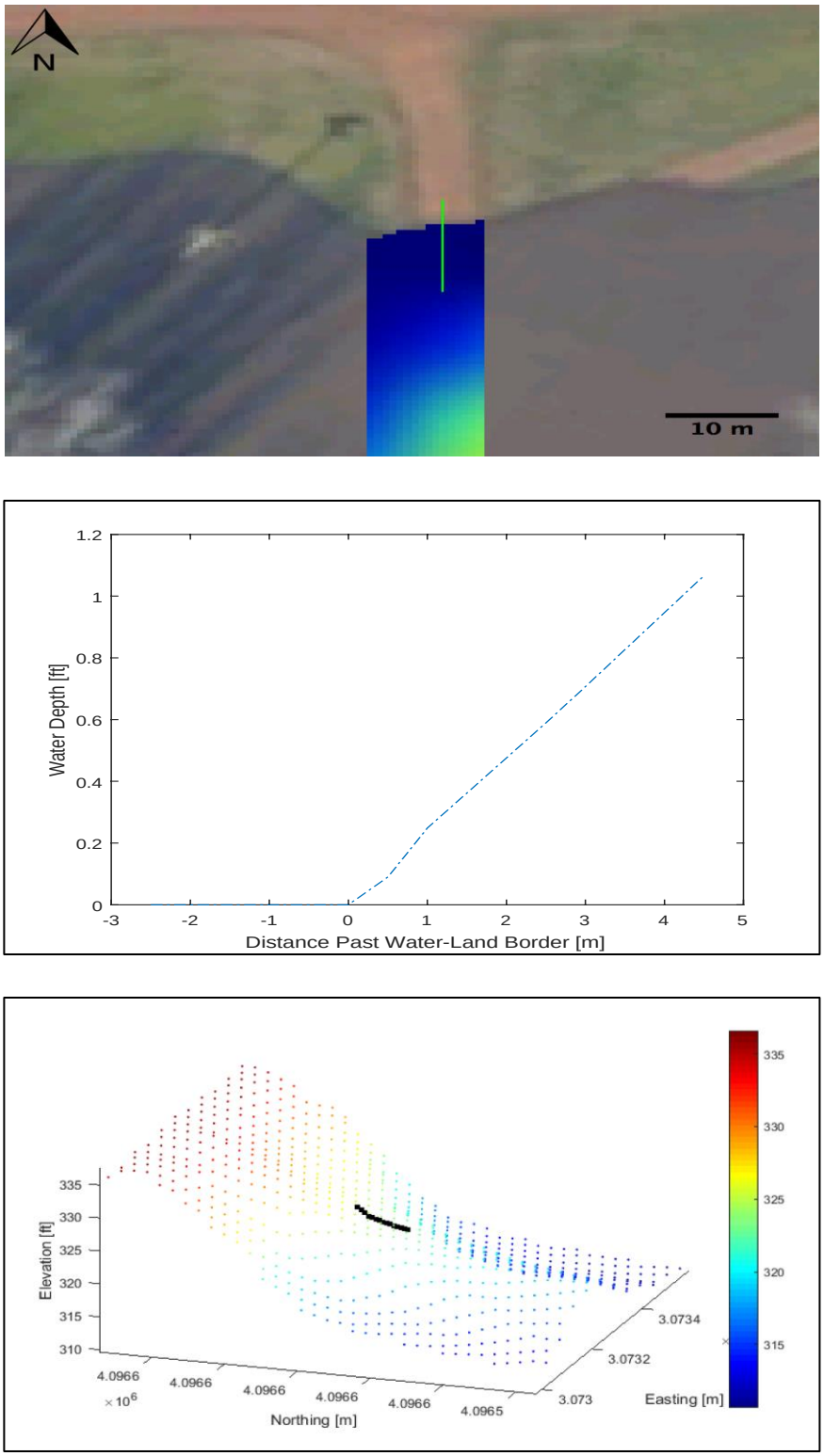


Figure 9: (Top) Roadway water depths near the water-land edge are overlaid on top of an image. (Middle) The depth profile of the green line from the top image. (Bottom) A 3-D view of the DEM with the same depth profile highlighted in black.

The three views in Figure 10 show what we were expecting to find at the borders between water and land. The water depth is zero before the border, is very shallow right past the border, and then follows the shape of the DEM. The pixels in the flood imagery are 0.5 m, so every one-meter tick mark in the plot represents two pixels. The water depths are about .2 ft (2.4 in) and .4 ft (4.8 in) at distances of two and four pixels from the border, respectively. Analysis of the five test scenes revealed similar results. Although this isn't a definitive method of validation, it shows that the algorithm is estimating water depths as expected.

5. Conclusions

This report details the methodology that allows debris volumes and water depths to be automatically estimated from the output of the detection algorithms. Alpha shapes are used to derive volume estimates from individual debris pile point clouds. A mean error of 6 m^3 was obtained through a validation experiment using real debris pile lidar data. Collecting data at a higher point density can significantly reduce the error, but even at low density the error is essentially insignificant (several debris piles in Haiti had volumes that were hundreds of cubic meters). Digital elevation model rasters are used to determine floodwater depth by finding the elevation where the water meets land. Although no validation data were available, results produced from testing the algorithm on real data were reasonable and exactly what one would expect. The outputs from these algorithms (debris volumes and water depths) are converted into shapefiles for easy viewing and integration with the Access Restoration module. Task Report 1.4 covers the details of the shapefile generation.

6. Deliverables

- Task 1.2 Report 1 (debris and flood detection algorithms)
- Task 1.2 Report 2 (volume and depth estimation algorithms)
- Standalone executable programs for volume and depth estimation

Two standalone programs were generated for this project. One application was created to handle all of debris detection, volume estimation, and debris shapefile output. The other application handles flood detection, depth estimation, and flood shapefile output. Rather than describe them here, a user manual was created for ease of use.

APPENDIX B:
Algorithms for Detection of Roadway Debris and Flooding

Access Restoration Project Task 1.2 Report 1 (of 2)
Algorithms for Detection of Roadway Debris and Flooding
Prepared by:
Digital Imaging and Remote Sensing Laboratory
Chester F. Carlson Center for Imaging Science

1. Overview

This report details the algorithms developed for automatically detecting roadway debris and flooding from commercial remote sensing (CRS) data. It is intended to give an idea of what is going on "under the hood" in the executable applications and the math/science that allow them to work.

The report is divided into five sections:

- Introduction: a brief discussion about the motivation for having software to automatically detect road blockages
- Methods: mathematical descriptions and examples of what each step of the algorithm is accomplishing (debris and flooding are discussed separately)
- Results: how the algorithms performed on test data sets
- Conclusions: a summary of the algorithms, results, and report
- Deliverables: a description of the deliverables for Task 1.2

2. Introduction

In the hours and days following a natural disaster, response efforts are typically focused on performing search and rescue missions, delivering relief to those who urgently need it, transporting the injured to medical facilities, and in some cases evacuating people, if there is still an immediate threat. All of these operations are dependent on unimpeded use of the road network. Obstructions to trafficability of these roads, such as large debris piles and floodwater, slow down response and recovery and therefore must be addressed immediately.

Remote sensing platforms provide a fast and safe way to gather overhead data of the entire region affected by the disaster. However, the data alone are not enough. Information needs to be extracted from the data – in this case the location of the blocked roadways. Typically, this requires a lot of time and manpower to meticulously look through the data, compare it to baseline data, and eventually identify obstructions. Such manual interpretation requires a lot of resources and is subjective – one observer might not agree with the findings of another.

Computer programs that can automatically turn raw CRS data into useful information products (blockage maps, optimal routes and restoration plans) help to alleviate the uncertainty and resource cost of manual methods. Although these programs are designed

to be simple "black boxes" for the user, understanding how and why they work can help the user to obtain the best results.

3. Methods

The debris and flood detection algorithms are very different – they use different data and mathematics to accomplish essentially the same task: detect roadway obstructions in the input data. Therefore, the methods will be split into a separate debris and flood section.

3.1 Debris Detection

The ultimate goal of the debris detection algorithm is to not only find occurrences of roadway debris piles, but to characterize them in terms of volume as well (to be addressed in the associated Report 2 for Task 1.2). Volume estimation requires 3-D data, so the decision was made to use airborne light detection and ranging (lidar) point clouds as the data input.

The algorithm needs two input files to run: an airborne lidar point cloud LAS file (*.las) and a shapefile containing the road network as a series of polygons (*.shp). Figure 1 shows an example point cloud and road network from Manhattan.

Figure 1: Airborne lidar point cloud and roadbed in Manhattan

3.1.1 Preprocessing

The point cloud and road network polygons need to be processed before detection is performed. The road network is truncated to the bounds of the point cloud and all points outside of the roadbed are discarded using the boundaries of the truncated road polygons. Point clouds often contain unwanted points (noise, points from birds, powerlines, etc.) that can affect operations and must be removed. Statistical outlier removal is used to remove these points. The statistical outlier removal stage operates by first calculating the mean Euclidean distance, $\bar{d}$, between each point, p, and its k closest neighbors. The mean (μ_k) and standard deviation (σ_k) of the distribution of mean distance are calculated, and points that have a mean Euclidean distance higher than multiple standard deviations above the distribution mean are classified as outliers and removed:

$$P^* = \{p \in P | \bar{d} \leq (\mu_k + \gamma \cdot \sigma_k)\}$$

where P^* is the entire point cloud after statistical outlier removal and γ is a scalar multiplier to control the severity of the point removal (lower γ = more points removed).

Following stastistical outlier removal, a digital elevation model (DEM) is generated. A DEM is a point cloud that contains only the points that represent the bare earth. In a case where ground points have been classified by the vendor, the DEM is computed by performing a Delaunay Triangulation on the known ground points using natural neighbor interpolation. If the point cloud does not have classified ground points, the DEM is produced using the Simple Morphological Filter (SMRF). The SMRF uses a linearly increasing window size, along with slope thresholding, to progressively remove objects from the bare earth until the DEM is achieved. Regardless of which method is used, the DEM grid spacing is set to μ_k. A normalized digital surface model (nDSM) is created by subtracting the point cloud from the DEM. The nDSM represents the height of all objects above the ground and is used later in the workflow for height thresholding. At this point the cloud is properly prepared for debris detection. Figure 2 shows a portion of a point cloud before and after preprocessing.

3.1.2 Point Filtering

It is challenging to define a single set of features to identify debris, because it varies remarkably in size, shape, and composition. Instead, individual points that are highly unlikely to be debris are first systematically removed using local surface properties and elevation cues. One of the main benefits of working with 3-D data is the ability to estimate the normal vectors of individual points, as well as the local distribution of normal vectors. These properties give insight to the underlying geometry of the surface that each point, *p*, belongs to. The normal vector of *p*

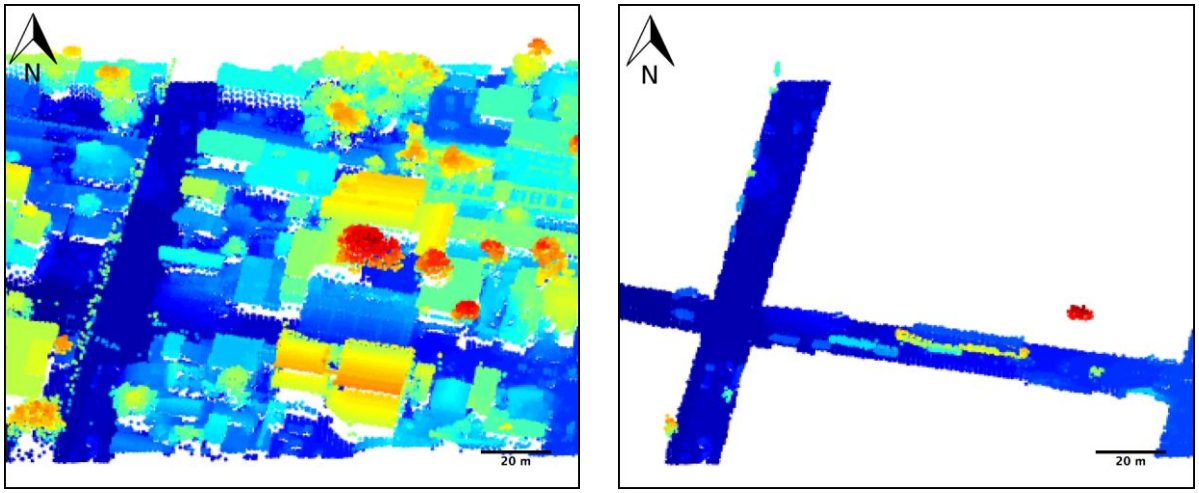


Figure 2: A point cloud before and after preprocessing.

can be calculated using all points in a local neighborhood, N_p, defined by the radius, r. For each query point, p, the neighborhood points are obtained using:

$$N_p = \{ q \in P | d(p, q) < r \}$$

where q are all of the points in the entire point cloud P and d is the Euclidean distance between two 3-D points. Eigen-analysis of the covariance matrix of N_p produces the eigenvalues $\lambda_1 < \lambda_2 < \lambda_3$. The eigenvector corresponding to λ_1 is the estimate of the point normal, $\vec{n}$. Airborne lidar data are collected from above, so the absolute value of the z-component of $\vec{n}$ is used to ensure the normal vector points outward from the surface.

The angle, θ, between each normal vector and horizontal is another useful property for point filtering:

$$\theta = 90° - \arccos(\vec{n}_z)$$

where $\vec{n}_z$ is the z-component of the normal vector. Points sampled from a horizontal surface have a θ of 90°. The neighborhood analysis is taken one step further by computing the distribution of normals in N_p. Eigenanalysis is again used, but this time on the covariance matrix of N_p resulting in $\lambda_1^n < \lambda_2^n < \lambda_3^n$. The eigenvalue corresponding to λ_2^n represents the maximum variation of normals on the Gaussian sphere and is useful for discerning points on a smooth surface (e.g., road, roof, etc.) from points on a rough surface (e.g., a debris pile).

A majority of the non-debris points can be identified and discarded using a combination of θ, λ_2^n, and height above ground, h. There are two sets of points that need to be removed in this stage: very low points (e.g., road, sidewalk, etc.) and points that are above the ground, but on very smooth surfaces (e.g., trucks, road barriers, etc). Low points are removed by finding those points that fall below a normal variation threshold (T_λ) and minimum height threshold (T_h). T_h is typically set to the vertical positioning accuracy of the system used to

collect the data (0.15 m for this research). Smooth points are identified by using a normal variation threshold that is 5x smaller without any height threshold. In both cases, a normal angle threshold (T_θ) is included to make sure that points on high-angle surfaces are not removed (these are much more likely to be debris).

$$I_1 = \{\, p \in P | (\lambda_2^n < T_\lambda) \wedge (h < T_h) \wedge (\theta > T_\theta) \}$$

$$I_2 = \left\{ p \in P \middle| \left(\lambda_2^n < \frac{T_\lambda}{5}\right) \wedge (h < T_h) \wedge (\theta > T_\theta) \right\}$$

where I_1 and I_2 are the indices of points classified as non-debris by the two equations above. The values of T_λ and T_θ are automatically derived from critical points in the histograms of λ_2^n and θ, respectively. T_h is set to the vertical point accuracy of the laser scanner. Figure 3 shows an image of a debris littered street along with the corresponding point cloud, before and after point filtering. A majority of the points remaining after filtering belong to debris, although some points on cars and small objects near the side of the road are still sometimes present.

3.1.3 Region Growing

In order to process and analyze debris piles at the object level, individual points belonging to the same pile must be clustered together. This is accomplished through a process called "*region growing*". A seed point is randomly chosen from the point cloud. The angle between the normal of the seed point and the normals of each of its neighbors, β, is computed.

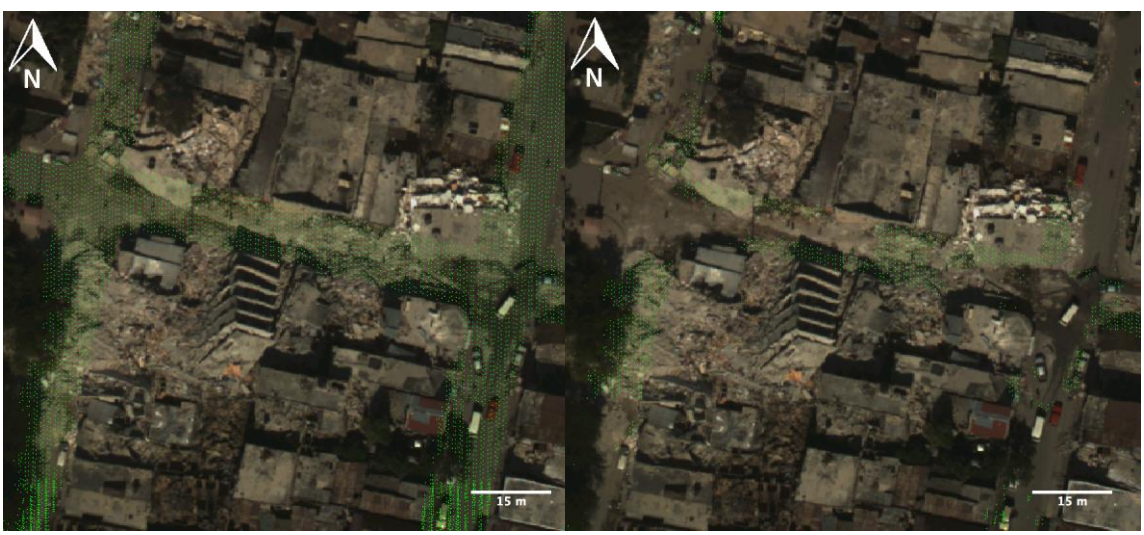


Figure 3: Non-debris points in a region affected by the earthquake are removed through point filtering. The points are overlaid on an aerial image for clarity

Any neighbor with a β below a certain threshold, T_β, is added to the region of the seed point. Through experimentation, a T_β, of 10° proved to be sufficient for preventing separate

objects from merging. Points are iteratively added to the region until no points can be added and then a new seed point is chosen. This process is repeated until all points have either been added to a region or discarded, because they couldn't be added to any region. A minimum region size of 15 points is used to discard regions that are too small to be significant. It should be noted that the minimum region size is only for the process of region growing. Debris piles are also discarded later on in the workflow after volume has been estimated by using a minimum volume threshold. Figure 4 shows the result of region growing being applied to a filtered point cloud.

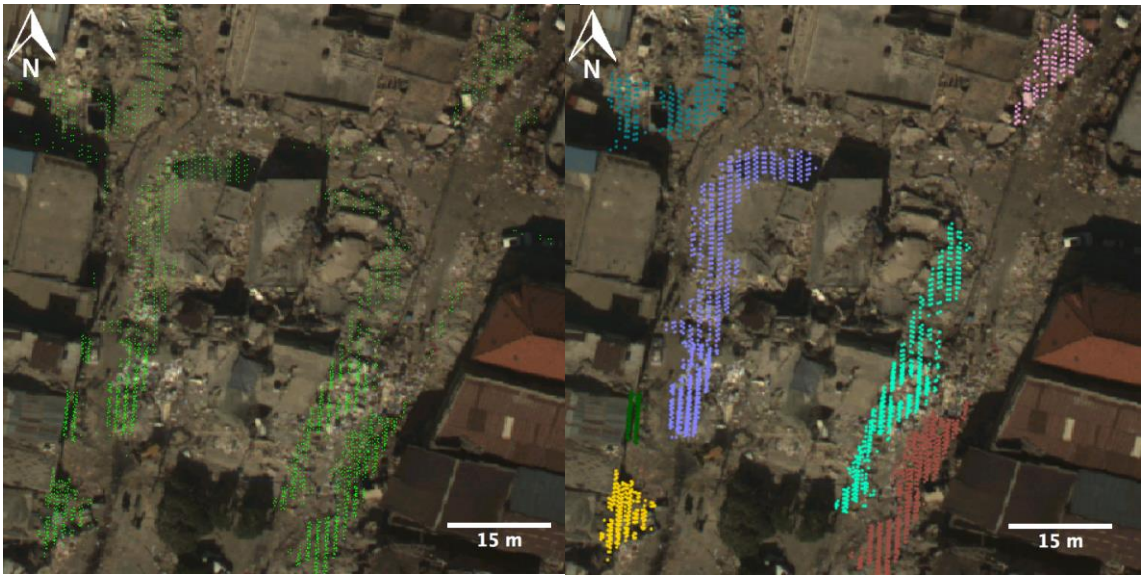


Figure 4: Region growing is applied to a set of filtered points in an area with several collapsed buildings. Each region is assigned a unique color.

3.1.4 Object Filtering

At this point, the initial set of objects (i.e., debris piles) have been detected. However, the results are likely to contain false alarms from small, non-debris objects such as small cars, road barriers, and medians. To filter out these objects, a metric called the height ratio (*HR)* is computed for each object. The height ratio is a measure of the portion of the object that is very low to the ground. Objects that have a *HR* greater than a threshold, T_{HR}, are discarded as non-debris objects.

$$HR = \frac{\#P_o < 2 \cdot T_H}{\#P_o}$$

where P_o are all of the points that make up an object (produced from region growing). Figure 5 shows a set of objects before and after object filtering.

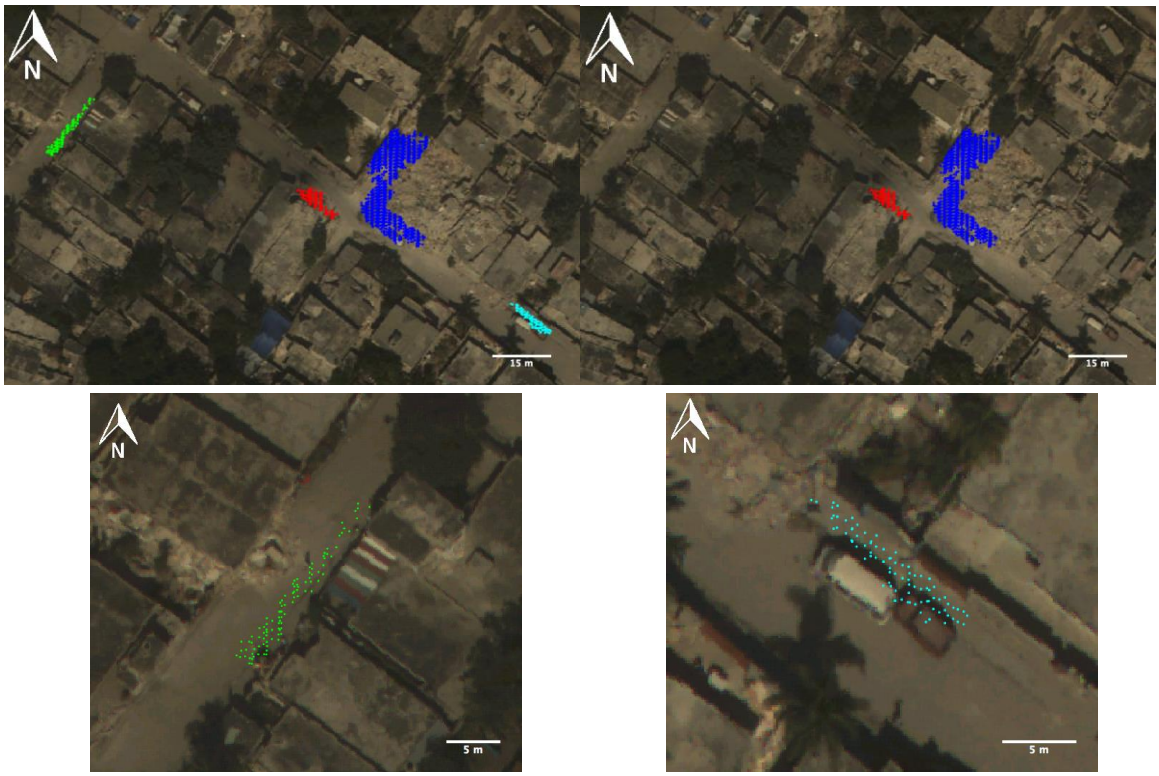


Figure 5: Object filtering is performed and removes two non-debris objects that were obtained from region growing. The two objects that are removed are a section of terrain and two vehicles parked very closely together that were mistaken for debris.

At this point, all of the debris piles have been detected and clustered as objects and the debris detection stage is therefore complete.

3.2 Flood Detection

The algorithm for detecting flooding in the roadways takes a GeoTIFF flood image (*.tif) and a road network shapefile (*.shp) and detects all of the pixels in the image that are believed to contain floodwater. It is highly recommended to use imagery that contains at least four spectral bands (red, green, blue, and near infrared), but the algorithm will run on three band imagery (red, green, and blue) or imagery with more than four bands (multispectral imagery). An example input flood image from the January 2016 flooding of the Mississippi River is displayed in Figure 6.

Figure 6: An image of a house surrounded by severe flooding from the Mississippi River. The road from the house to the nearest street is completely flooded.

3.2.1 Finding a Seed Pixel

The premise behind the whole algorithm is that if one flood pixel can be obtained, then spectral information about that pixel can be used to find the rest of the flood pixels in the image. The initial flood pixel is referred to as a "*seed pixel*". The algorithm was constructed such that the user has the option to pick the seed pixel(s) (a foolproof method of guaranteeing a true flood pixel is selected) or let the algorithm attempt to automatically find the seed pixel (advantageous if you need to run the algorithm on several images). Additionally, the user can select the seed pixel for one image and then use that pixel to process several images. The rest of the section will describe the automatic method of finding the seed pixel.

Apart from very shallow water, floodwaters tend to look very homogeneous from overhead imagery (as can be seen in Figure 6). To find a seed pixel, the algorithm tries to find a pixel at the center of the most homogeneous region of the image.

First, the image is converted from the RGB color space to the HSV color space. The HSV, or hue-saturation-value, color space. The hue channel is a good single band representation of the uniformity of the water. Figure 7 shows an image in RGB and the corresponding hue band.

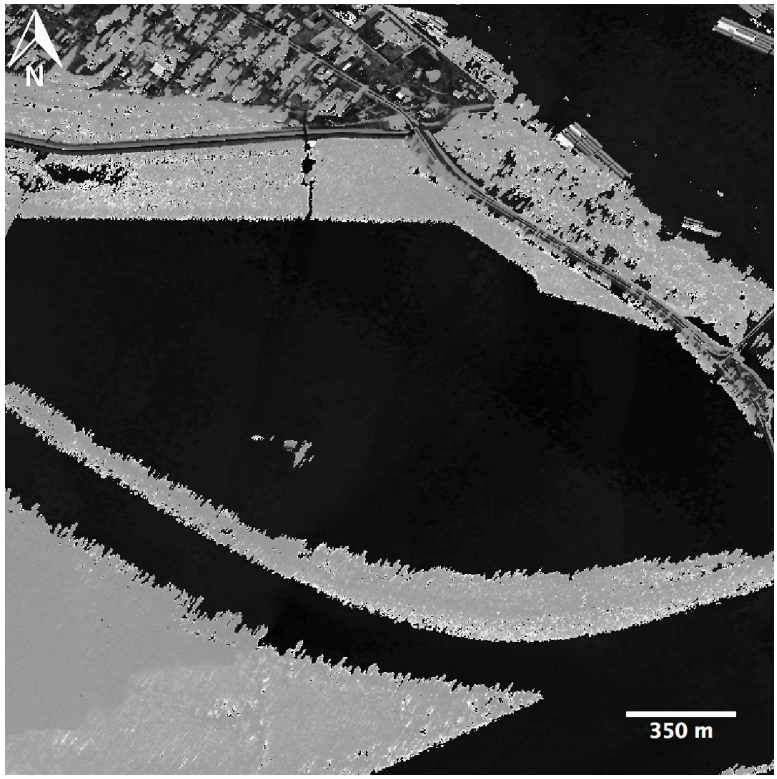


Figure 7: The hue channel from the HSV color space of the flood image from Figure 6. The water appears very uniform in the hue channel.

An entropy image of the hue band is then calculated. Entropy is a statistical measure of randomness and is good for analyzing texture. The entropy of each pixel is calculated using a 9x9 window of pixels around it. The lower the entropy of a pixel is, the more homogeneous the region it sits in is. This makes it a good metric for detecting water regions.

The entropy image is then converted into a binary image using Otsu's method. Otsu's method creates a binary image from a grayscale image by finding a threshold that maximizes the variance between two classes while simultaneously minimizing the variance within the classes. In this case, the two classes are high entropy regions and low entropy regions. Small areas in the resulting binary image are removed by performing a morphological opening with a disk-shaped structuring element with radius r_{se}, which is typically set to 5 pixels. Finally, connected components with less than 500 pixels are removed in an operation known as area opening. At this point, the remaining binary image contains a set of homogeneous regions. However, not all of these homogeneous regions belong to water. Figure 8 shows the set of homogeneous regions for the flood image in Figure 6.

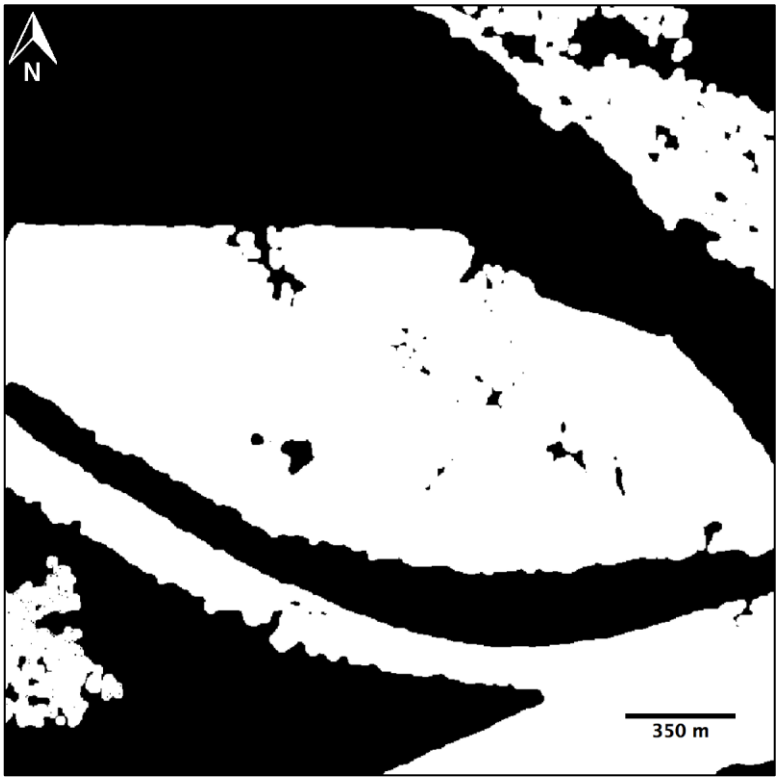


Figure 8: The set of homogeneous regions produced from Otsu's method, morphological opening, and area opening.

The next step is to remove regions that are unlikely to be water. These regions can be grass fields, shadows, asphalt, or any other large homogeneous area. Two masks are created to remove the erroneous regions: a dark mask and a vegetation mask. The dark mask is created by finding very dark pixels (i.e., shadows, asphalt, etc.), and the vegetation mask is created using either the Normalized Difference Vegetation Index (NDVI) if the image is 4 bands, or by finding green pixels if the image is 3 bands:

$$M_{dark} = (R < .05) \wedge (G < .05) \wedge (B < .05),$$

$$M_{veg4} = NDVI < -.25,$$

$$M_{veg3} = (G > R) \wedge (G > B),$$

$$NDVI = \frac{NIR - RED}{NIR + RED}$$

where M_{dark}, M_{veg4}, and M_{veg3} are the dark mask, 4-band vegetation mask, and 3-band vegetation mask, respectively. R, G, and B are the red, green, and blue bands of the image.

At this point, the remaining regions are assumed to belong to floodwater. An assumption is made that the most "flood-like" pixel will be in the middle of a homogeneous region. Therefore, the Euclidean distance transform image is calculated and the pixel with the highest value is selected as the seed pixel. Figure 9 shows the distance transform image and the resulting seed pixel. As mentioned before, the seed pixel(s) can also be chosen manually. If there is no flooding present in the image, the automatic process will still choose a seed pixel from the most homogeneous region.

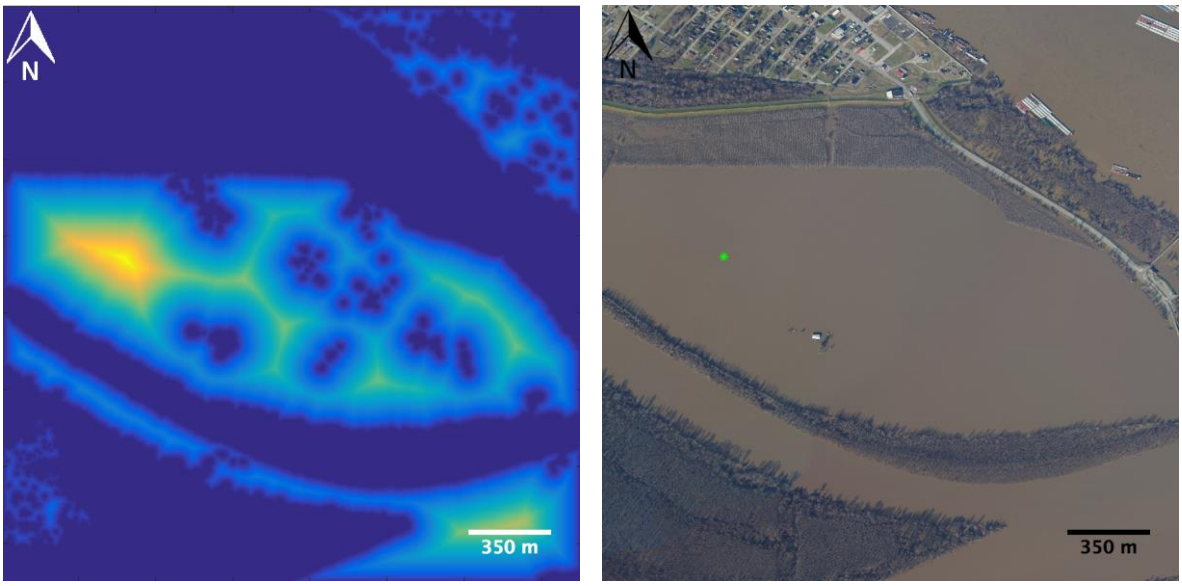


Figure 9: (Left) The Euclidean Distance transform is used to find the pixel the farthest from the edge (yellow = larger distance). (Right) The chosen seed pixel is marked with a green star.

3.2.2 Detecting all Flood Pixels

Once a seed pixel is automatically detected or selected, it is used to detect similar pixels (i.e., flood pixels) in the image. First, spectral angle mapper (SAM) is used to find pixels that are very similar to the seed pixel. SAM is a target detection method that works by calculating the angle between the spectral vector of the seed pixel and the spectral vector of all other pixels in the image. Pixels with spectral angles below a certain threshold, T_{SAM}, are stored for use in the next step. T_{SAM} is set to 0.075 in practice. The equation for calculating the SAM statistic is:

$$r_{SAM}(\boldsymbol{x}) = -\cos^{-1}\left(\frac{\boldsymbol{s}^T\boldsymbol{x}}{\sqrt{(\boldsymbol{s}^T\boldsymbol{s})(\boldsymbol{x}^T\boldsymbol{x})}}\right)$$

where $\boldsymbol{s}$ is the spectral vector of the seed pixel (or the mean of all seed pixels if there are multiple) and $\boldsymbol{x}$ is the spectral vector of a test pixel.

There is some variation in the spectra of flood pixels due to things like water flow and water depth. To try to detect all of the flood pixels despite these variations, another target detection method called the Quadratic Matched Filter (QMF). The QMF takes into

account the variation of both the background and the target and is good at mitigating false alarms. Typically the QMF would also take into account noise, but it was decided to ignore this term for simplicity in the implemented algorithm. Pixels with a QMF detection statistic, r_{QMF}, greater than a certain threshold, T_{QMF}, are the detected flood pixels. T_{QMF} was set to 80 during testing. The QMF equation is:

$$r_{QMF}(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{\mu_b})^T {\Sigma_b}^{-1} (\boldsymbol{x} - \boldsymbol{\mu_b}) - (\boldsymbol{x} - \boldsymbol{\mu_s})^T {\Sigma_s}^{-1} (\boldsymbol{x} - \boldsymbol{\mu_s}) + \mathbf{log} \frac{|\Sigma_b|}{|\Sigma_s|}$$

where $\boldsymbol{\mu_b}$ is the mean spectrum of the background (in this case, the entire image), $\boldsymbol{\mu_s}$ is the mean spectrum of the target pixels (the pixels detected from SAM), Σ_b is the covariance matrix of the background, and Σ_s is the covariance matrix of the target pixels. Figure 10 shows the results of QMF on the flood image from Figure 6.

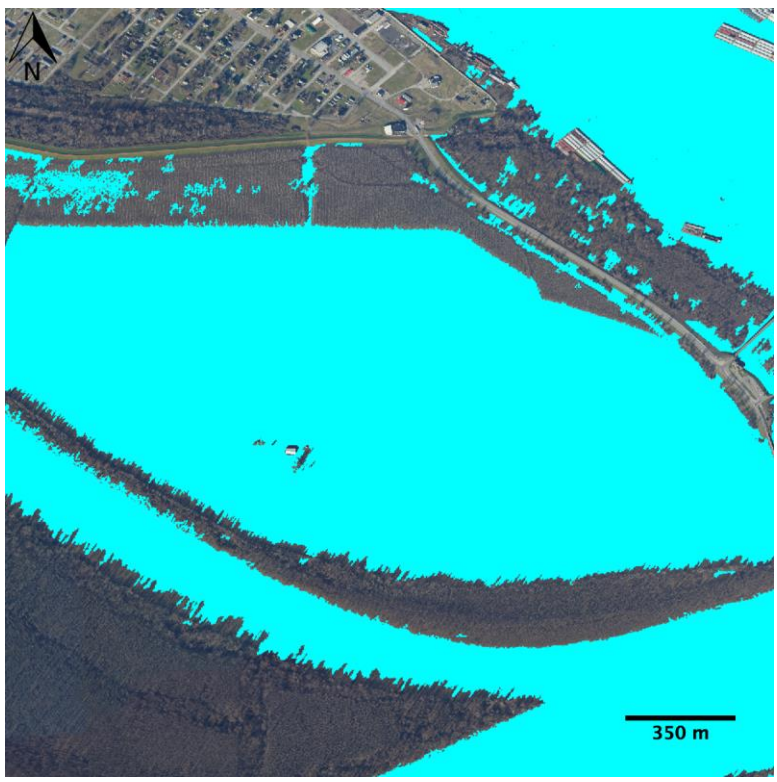


Figure 10: The flood pixels detected by the QMF are shown in cyan.

At this point, all the flood pixels in the entire image have been detected. The goal is to find flood pixels in the roadways, so the detection can then be segmented to the roadways by simply multiplying by a road mask. The road mask is calculated by converting the road network shapefile into a road mask the same size as the flood image. Figure 11

shows the roadway flood pixels as well as a pre-flood image of the same scene for comparison.

Figure 11: (Left) A pre-flood image of the scene from Google Maps. (Right) The detected roadway flooding pixels

4. Results

In order to understand the performance capabilities of the algorithms, it is vital to test them on datasets for which some reference of what the expected results should be has been created.

4.1 Debris Results

Reference data to validate the debris detection results for seven scenes in Haiti were created by manually tracing every pile visible in the high-resolution (0.15m) imagery that was simultaneously captured with the lidar data. The bounding polygons of the 2-D projections of the detected debris piles are intersected with the set of validation polygons for each scene. If a debris pile polygon detected by the algorithm overlaps with a validation polygon, then it is classified as a true positive (*TP*). If there is no overlap between the detected debris polygon and any of the validation polygons, then it is labeled as a false positive (*FP*). If a validation polygon does not overlap any of the detected debris polygons, then it is classified as a false negative (*FN*). The validation polygons are sometimes larger than the detected polygons because debris that was present in the nadir aerial imagery is very low to the ground and discarded during the point filtering stage of debris detection. The low points are intentionally discarded as insignificant contributions to the overall debris pile. Therefore, no threshold is placed on the amount of overlap between the two sets of polygons to achieve a correct detection. During

validation, if a digitized pile was found to be completely below the minimum height threshold, it was deleted and the results were recalculated. The *TP*, *FP* and *FN* counts are used to characterize the performance in terms of completeness, correctness, and quality. Completeness reflects the percentage of validation debris piles that are detected by the algorithm. Correctness reflects the percentage of detections that are true debris piles. Quality is a measure of overall performance that takes into account both the completeness and the correctness of the results. The results for all seven of the scenes are displayed in Table 1.

$$Completeness = \frac{TP}{TP + FN} \times 100$$

$$Correctness = \frac{TP}{TP + FP} \times 100$$

$$Quality = \frac{TP}{TP + FP + FN} \times 100$$

Table 1: Debris detection results for the seven test sites.

Site	**# Validation Piles**	**TP**	**FP**	**FN**	**Completeness [%]**	**Correctness [%]**	**Quality [%]**
1	21	24	3	0	100.00	88.89	88.89
2	5	5	0	0	100.00	100.00	100.00
3	14	13	0	0	100.00	100.00	100.00
4	9	8	1	1	88.89	88.89	80.00
5	11	11	3	0	100.00	78.57	78.57
6	7	7	1	0	100.00	87.50	87.50
7	21	24	3	1	96.00	88.89	85.71
All	88	92	11	2	**97.87**	**89.32**	**87.62**

Figure 12: Validation debris pile polygons (red) and polygons of the debris piles detected by the algorithm (teal) are overlaid on aerial imagery.

The proposed algorithm achieved an overall quality score of approximately 88% when tested on seven scenes (containing a total of 88 validation debris piles), indicating that it is capable of satisfactory performance in automated roadway debris pile detection. Even more importantly, the algorithm achieved a completeness score of approximately 98%. The

completeness metric penalizes false negatives, which in the case of debris detection for emergency response, are arguably more costly than false positives. A false negative could result in an ambulance that is transporting critically injured survivors to take a route to the hospital that is blocked by debris. The ambulance would have to turn around and attempt to find a new route, preventing the injured from getting timely treatment and possibly even resulting in death. Of the 88 validation debris piles, only two were missed, and no more than one pile was missed in a single scene.

The eleven false positives that were detected were the result of large vehicles or clusters of small vehicles being mistaken for debris. These false positives are much smaller than the actual debris piles produced from collapsed buildings and could be alleviated by using a higher minimum volume threshold ($20m^3$ was used during testing) or using lidar data with a higher point density.

4.2 Flooding Results

The goal of the flood detection algorithm is to detect flooded roadways. To assess the results of the algorithm, bounding polygons of flood pixels within road boundaries were manually digitized using GIS software. The flood algorithm was tested on five flood images and a pixel-to-pixel comparison was performed using the output and the validation polygons. The same metrics that were used for the debris detection validation (TP, FP, FN, completeness, correctness, and quality) were used for flooding as well.

Overall, the flood detection algorithm was successful in detecting roadway flood pixels, achieving a quality score of 92% across the five scenes. Completeness and correctness scores of approximately 95% and 97%, respectively, suggest that false negatives and false positives are minimal. Figures 13 and 14 show the detection results for the first two scenes, along with some examples of false positives and false negatives.

Table 2: Roadway flood pixel detection results for five flood images

Image ID	# Flood Pixels	TP	FP	FN	Completeness [%]	Correctness [%]	Quality [%]
flood1	9,1297	83,156	6,002	8,141	91.08	93.27	85.46
flood2	125,444	122,698	926	2,746	97.81	99.25	97.09
flood3	33,171	30,687	1,415	2,484	92.51	95.59	88.73
flood4	37,862	36,067	115	1,795	95.26	99.68	94.97
flood5	16,836	16,307	976	529	96.86	94.35	91.55
All	304,610	288,915	9,434	15,695	**94.85**	**96.84**	**92.00**

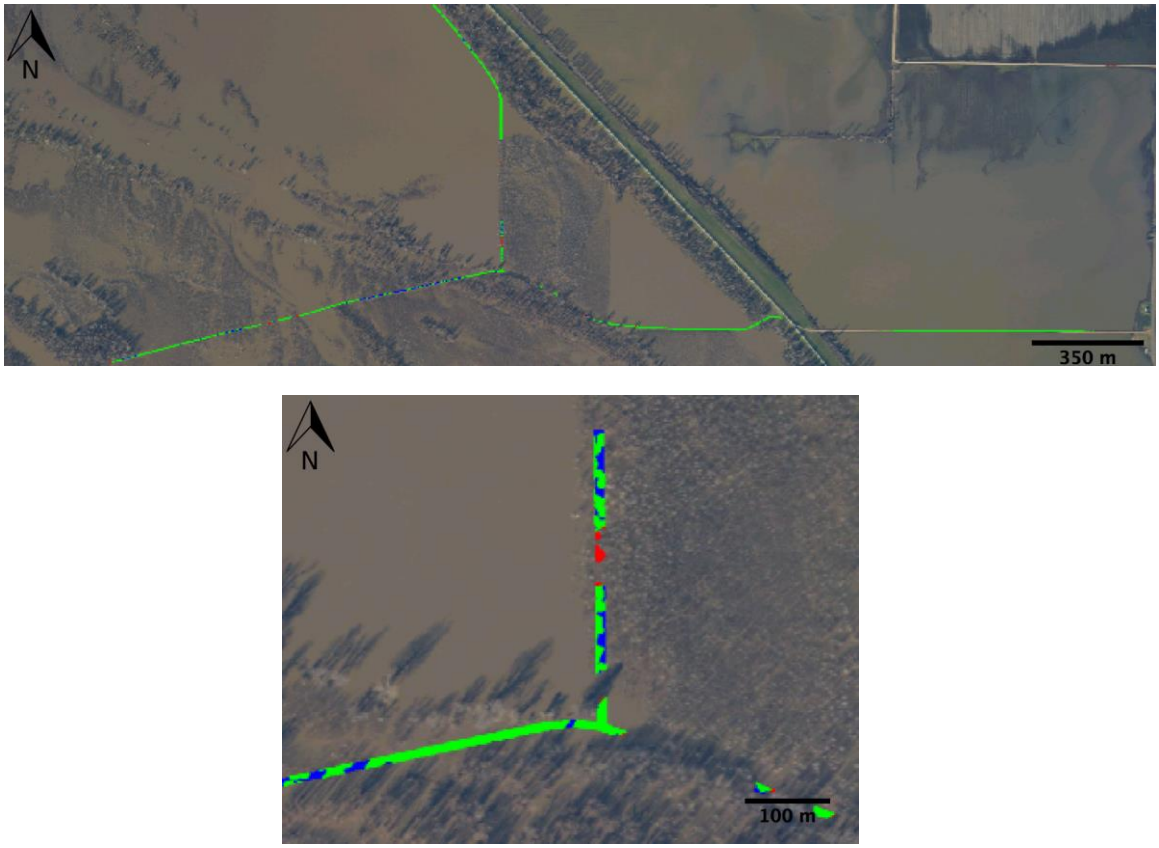


Figure 13: (Top) Roadway flood detection results for scene 1. Green = TP, red = FP, blue = FN. (Bottom) A zoom view reveals that most FPs and FNs are caused by shadows, trees, etc.

Figure 13 displays a detection result image for the first flooding scene. The image is color coded such that a green pixel indicates a true positive, a red pixel indicates a false positive, and a blue pixel indicates a false negative. The false positives for this image mostly appear in pixels amongst areas with a lot of trees. It is likely that these false positives are actually true flooding pixels that were neglected when the validation set was created. It is extremely meticulous to label every pixel, especially when they are individual or small groups of pixels inbetween trees. A majority of the false negatives are caused by shadows cast by trees across the roadways. The shadows affect the spectra of the pixels in regions that are actually flooded, preventing them from being detected. Additional spectral bands in the shortwave infrared (SWIR) could help to prevent shadowed flood pixels from being neglected. Despite the small amount of false positives and false negatives, a vast majority of the roadway flood pixels are detected and can be passed on to the depth estimation stage of the algorithm.

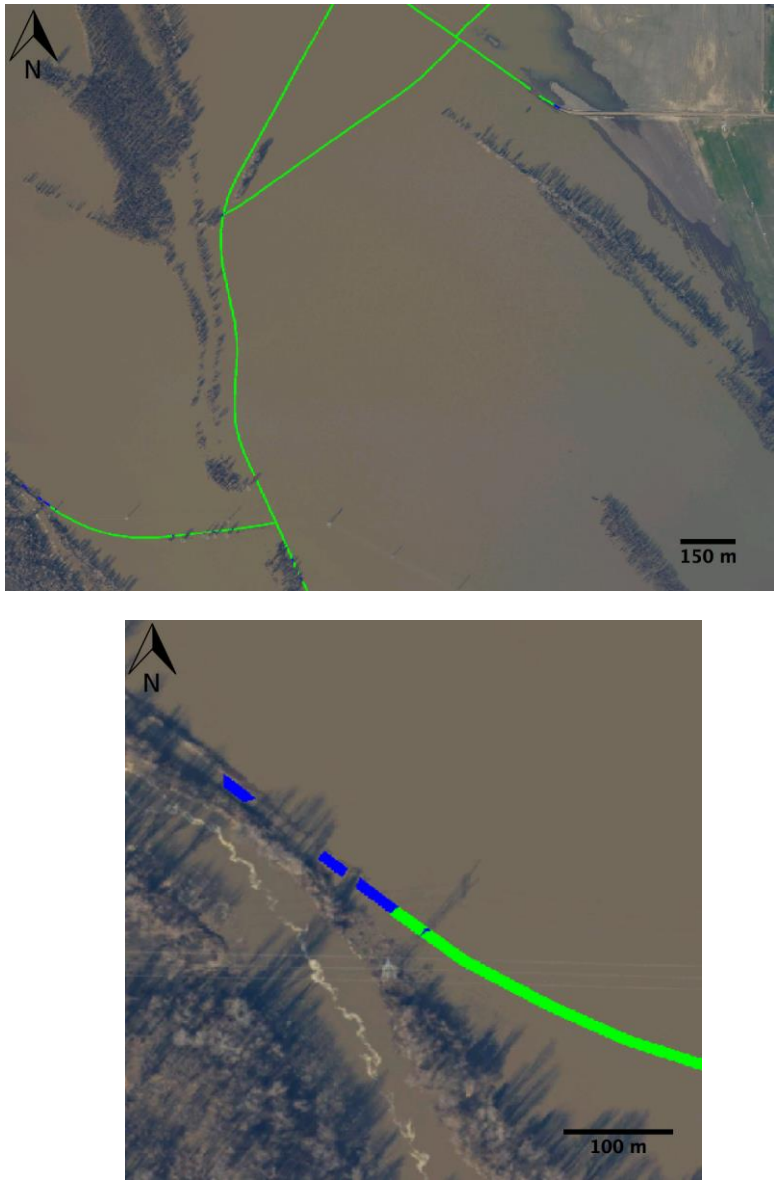


Figure 14: (Top) Roadway flood detection results for scene 2. Green = TP, red = FP, blue = FN. (Bottom) A zoom view reveals that most FNs are caused by small flood regions that were removed during the morphological and area openings

Scene two (Figure 14) contained almost no false positives. Similar to scene one, the majority of the false negatives are a result of shadow regions. In some cases, the shadow regions are detected by the QMF, but so small because they are isolated from the rest of the pixels that they are accidentally removed during the area opening stage. Given the 97% completeness and 98% correctness for this image, it is safe to say that accurate information products can be derived from the detected roadway flood pixels.

5. Conclusion

This report details the mathematics happening at each step of the roadway debris and flooding detection algorithms. The algorithms use road network shapefiles and CRS data (lidar for debris, imagery for flooding) to automatically detect obstructions to the roadway. By testing the debris detection algorithm on seven point clouds from the 2010 earthquake in Haiti, we have shown that nearly 98% of all debris can be detected along with an overall quality of performance of over 86%. Using 4-band flooding imagery from the January 2016 Mississippi River flooding we have shown that the flood detection algorithm was able to detect 95% of roadway flood pixels while achieving an overall quality of 92%. Similar results can be expected when these algorithms are applied to other post-disaster lidar data or flood imagery. Although there are many different types of algorithms that could be applied to these types of CRS data, the proposed algorithms were chosen because they provide accurate and reliable results. The detection outputs feed directly into the volume and depth estimation algorithms, which are described in Task 1.2 - Report 2.

6. Deliverables

- Task 1.2 Report 1 (debris and flood detection algorithms)
- Task 1.2 Report 2 (volume and depth estimation algorithms)
- Standalone executable programs for debris and flood detection

Two standalone programs were generated for this project. One application was created to handle all of debris detection, volume estimation, and debris shapefile output. The other application handles flood detection, depth estimation, and flood shapefile output. Rather than describe them here, a user manual was created for ease of use.

APPENDIX C:
Generating GIS Products from Detection and Characterization Algorithm Outputs

Access Restoration Project Task 1.4 Report
Generating GIS Products from Detection and Characterization Algorithm Outputs
Prepared by:
Digital Imaging and Remote Sensing Laboratory
Chester F. Carlson Center for Imaging Science

1. Overview

This report details the methodology used to take the outputs from the detection (debris and flood) and estimation (volume and depth) algorithms and packages them into map-ready GIS products that can be viewed as standalone information products and also passed into RPI's ARP module.

The report is broken into four sections:

- Introduction: a brief discussion about the motivation for generating shapefiles from the detection and estimation modules
- Shapefiles: descriptions of how the characterized road obstructions are converted into shapefiles
- Conclusions: a summary of the shapefile conversion process and outlook on how they can be used as information products
- Deliverables: a description of the deliverables for Task 1.4

2. Introduction

The detection and estimation algorithms detailed in Task 1.2 - Reports 1 & 2 are able to ingest raw CRS data and automatically detect debris piles, flooded roadways, and estimate their volumes and depths, respectively. Although some functionality is included in the standalone applications to save output images such as the flood detection image and the water depth image, the true utility of the algorithms is the ability to automatically convert the results into a GIS-ready shapefile.

The shapefiles provide a convenient and comprehensive way of viewing the spatial location and extent of debris piles and flooding, while also embedding attributes within the polygon of each obstruction (volume, passable width, depth, etc.). Additionally, the shapefiles provide the data necessary to seamlessly integrate with the Access Restoration module, which provides further visualization and allows for restoration times to be computed.

3. Methods

3.1 Debris Shapefiles

Two shapefiles are produced when running the debris application. The first shapefile is the debris shapefile. The debris shapefile contains a set of polygons. Each polygon represents a debris pile and contain the following attributes: LocalID, Volume, and PassWidth. Figure 1 shows a screenshot from ArcGIS zoomed into one of the debris polygons for a scene in

Haiti. The information on the box shows the attributes for the red debris polygon in the center of the image.

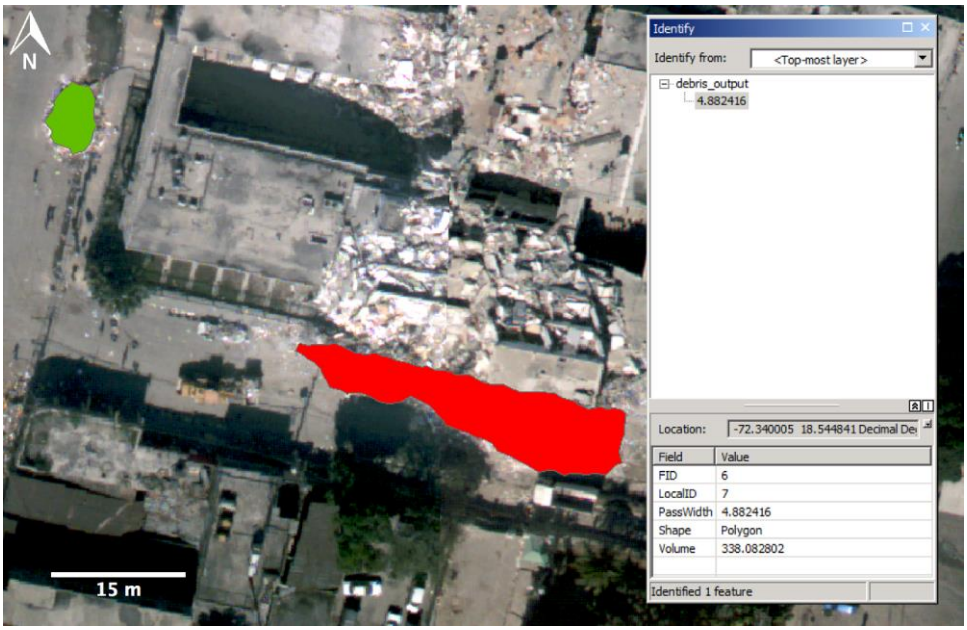


Figure 1: A screenshot from ArcGIS shows a close up view of what the debris polygons look like in a GIS viewer. The polygons are colored by volume (green = less volume, red = more volume). Clicking the red polygon reveals its attributes, such as passable width (in meters) and volume (in m^3).

The ***LocalID*** attribute is used for internal debugging and will likely be removed in future releases once integration with RPI has been completed.

The ***Volume*** attribute represents the estimated volume of the debris pile in m^3 and is computed by the algorithm presented in Task 1.2 - Report 2. In Figure 1, the volume attribute is used to color the color the debris pile (the lowest volumes are in green and the largest are in red).

The ***PassWidth*** attribute stands for "passable width" and represents the amount of room that a vehicle has to pass through the road because of the debris pile. The passable width is calculated based on the distances between the side of the road and other adjacent debris piles. Figure 2 shows an example of the passable width for two debris piles in a roadway.

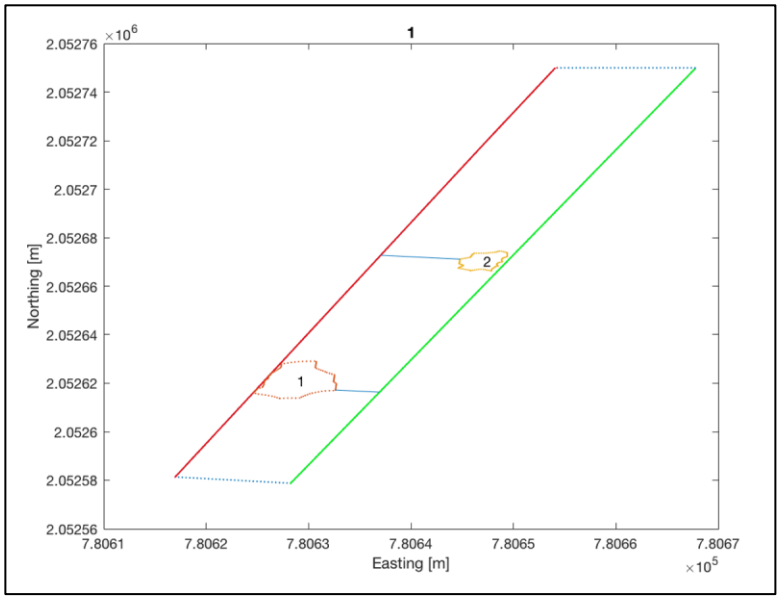


Figure 2: An example demonstrating the passable width attribute. Two debris piles are shown on a roadway (opposing road edges are marked in red and green). The blue line illustrates the passable width due to the debris pile and road edge.

The road shapefile is a set of polygons of roads that contain debris. The polygons come directly from the input road network shapefile (only for roads that contain debris, the others are excluded). Each road polygon contains a LocalID, PassWidth, and TotalVol attribute. Figure 3 shows a screenshot from ArcGIS of road polygons and the attributes for one road.

The ***LocalID*** attribute is used for internal debugging and will likely be removed in future releases once integration with RPI has been completed.

The ***TotalVol*** attribute is the sum of the individual volumes of all debris piles located within the boundaries of the road polygon. The total volume is expressed in terms of m^3.

The ***PassWidth*** attribute represents the passable width of the entire road. This value is calculated by taking the minimum passable width for all debris piles within the road. The minimum passable width of all debris piles is the limiting factor for a vehicle to travel from one end of the road to the other.

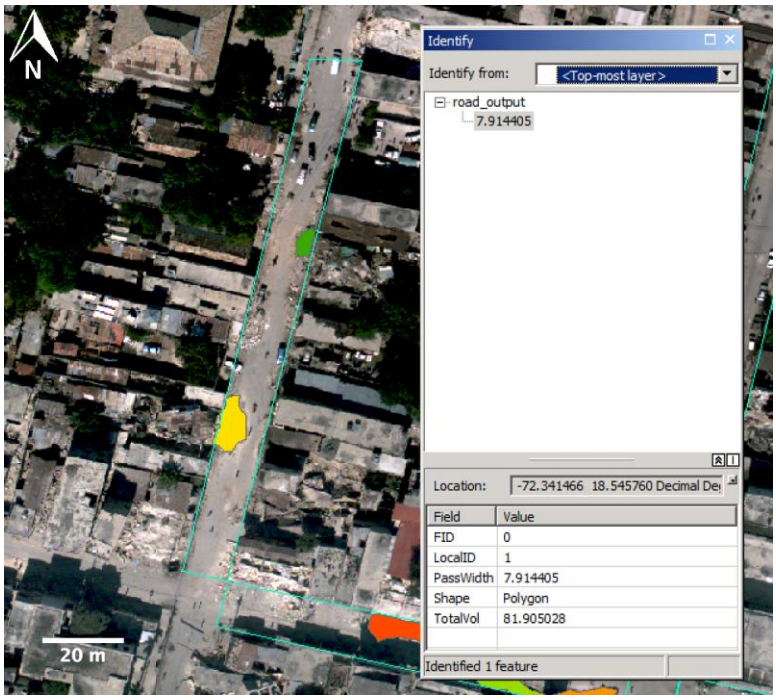


Figure 3: A screenshot from ArcGIS shows the road polygons (outlined in teal) as well as the debris piles. The information box shows the attributes for the given road such as passable width and total volume of debris on the road.

3.2 Flooding Shapefiles

The flooding application produces a single shapefile. This shapefile contains polygons that represent areas of approximately equivalent water depth in the roadway. Figure 4 shows a screenshot from ArcGIS showing the flood polygons from a flooded farm near the Mississippi River.

Figure 4: An ArcGIS screenshot of roadway flood polygons overlaid on a flooding image from the Mississippi River. The polygons are colored by depth (green = shallow, red = deep). The information box shows attributes such as depth and area for a single polygon.

The flooding shapefile contains two attributes for each polygon: Depth and Area. The depth attribute is the approximate depth of water throughout the polygon (rounded to the nearest foot) and the area of the polygon (in ft^2).

As part of the depth estimation algorithm (outlined in Task 1.2 - Report 2), a depth raster is produced for roadway flood pixels. The raster represents the water depth at each of the pixels. As you could imagine, it would be extremely inefficient and memory intensive to produce a separate shapefile for each pixel. Instead, pixels are rounded to the nearest foot and then pixels with the same depth are combined to form polygons. These are the polygons you see in Figure 4. The areas of the polygons are computed using the known pixel sizes. Figure 5 shows a section a depth image for a section of flooded roadway before and after rounding. The image on the right shows how the rounding creates regions of equal depth, which then become the polygons in the shapefile.

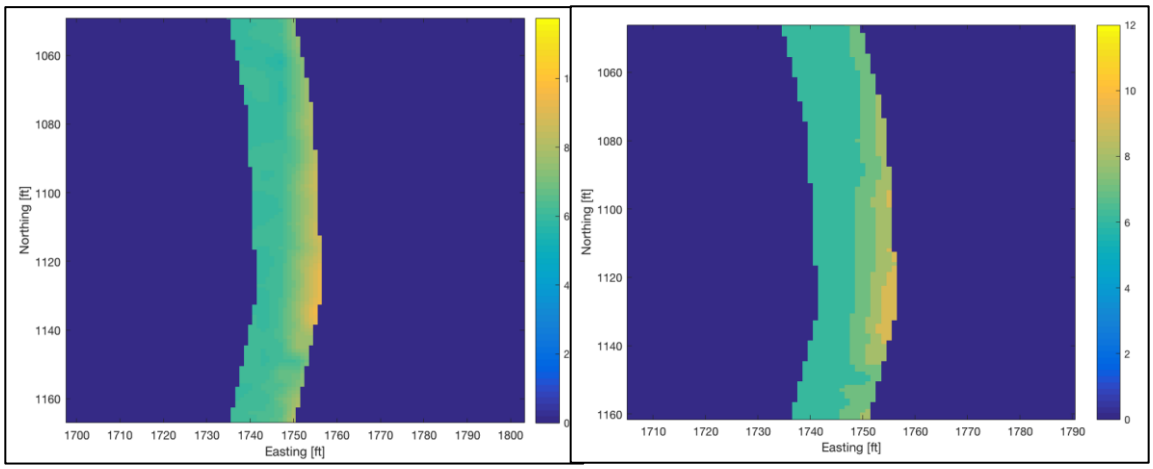


Figure 5: (Left) Depths for a flooded section of roadway before rounding; each pixel represents a decimal depth. (Right) Depths for a flooded section of roadway after rounding. Regions of equal depth form after rounding, which then become polygons in the shapefile.

4. Conclusions

Shapefiles provide an easy, universal way to visualize the results of the depth and flood detection and estimation algorithms. The shapefiles can be loaded into any GIS program and are automatically placed in the right location regardless of projection. The shapefiles can be placed on top of imagery or base maps (such as OpenStreetMap, USGS, etc.) to view the debris piles or flooded roadways in context. Each debris, road, or flood polygon contains characteristics such as debris volume, passable width, water depth, and area. Along with being convenient for visualization and analysis, the shapefiles can feed directly into RPI's algorithms for calculation of restoration times.

5. Deliverables

- Task 1.4 Report
- Standalone executable programs for volume and depth estimation

APPENDIX D:
Technical Advisory Committee Meeting Slides

Decision Support System for Optimal Access Restoration in Post Disaster Environments -Technical Advisory Committee (TAC) Meeting- January 2016

Presenters:

José Holguín-Veras,

William H. Hart Professor,
Director of the Center for Infrastructure, Transportation, and the Environment
Rensselaer Polytechnic Institute
jhv@rpi.edu

Jan van Aardt,

Associate Professor,
Acting Director, Digital Imaging and Remote Sensing Group, Chester F. Carlson Center for Imaging Science,
Rochester Institute of Technology
vanaardt@cis.rit.edu

Outline of Webinar

- Overview of TAC and team members
- Description of the problem we are trying to address
- Vision and Goals
- Project Overview
  - Task descriptions
  - Status
  - Next steps
- Discussion

Center for Infrastructure, Transportation, and the Environment

Overview of Team Members and TAC

Center for Infrastructure, Transportation, and the Environment

PI and Co-PI

- **Dr. José Holguín-Veras**

  William H. Hart Professor, and Director of the Center for Infrastructure, Transportation, and the Environment; and the Volvo Research and Educational Foundations' Center of Excellence on Sustainable Urban Freight Systems at the Rensselaer Polytechnic Institute. His research group has pioneered the multidisciplinary study of post-disaster humanitarian logistic operations, using a holistic approach encompassing field work, quantitative characterization of operations, and basic research on mathematical modeling.

- **Dr. Jan van Aardt**

  Associate Professor at Rochester Institute of Technology, and Chester F. Carlson Center for Imaging Science. He has received large disaster response research and development awards and routinely interacts with state and federal emergency response agencies, delivering CRS disaster response products. He holds PhD and MS degrees from Virginia Tech University, and BSc Don Forestry from University of Stellenbosch, South Africa.

Technical Advisory Committee Members

- **Mrs. Kathy Fulton**

  Executive Director at American Logistics Aid Network: organization comprised of hundreds of supply-chain businesses who stand poised to respond in the event of disasters. Mrs. Fulton graduated Summa Cum Laude from University of South Florida with Dual Masters' degrees in Business Administration and Management Information Systems; also holds a B.S. in Mathematics from Northwestern State University.

- **Mr. Jon Meyer**

  Experienced transportation professional, with expertise in logistics, transportation and distribution. This includes a twenty-seven year career at CSX Transportation. Retired Colonel in the Air Force Reserve, served as a logistics staff officer on the Air Staff, The Pentagon. Awards include the Legion of Merit, Meritorious Service Medal, and Air Force Commendation Medal. From 2012-2014 was chair for TRB's task force on Logistics of Disaster Response and Business Continuity.

Technical Advisory Committee Members

- **Mr. Phillip Palin**

  Principal investigator for supply chain research at the Institute for Public Research, he is also a senior fellow in homeland security at Rutgers University Graduate School, and staff consultant for supply chain resilience at the National Academy of Sciences. Author of Considering Catastrophe, and principal author of Catastrophe Preparation and Prevention series from McGraw-Hill, among other publications on the subject.

- **Dr. Chris Renschler**

  Associate Professor of Geography and Co-Founder and of the Center for GeoHazards Studies at the University at Buffalo. Since 2010, is faculty member of Multidisciplinary Center for Earthquake Engineering Research. His research interests are Integrated Environmental Management, GIScience, Environmental Modeling and Extreme Events. Holds a Ph.D. in Natural Sciences from University of Bonn, Germany; B.Sc. and M.Sc. degrees in Geoecology from Technical University of Braunschweig

Technical Advisory Committee Members

- **Dr. Jie Shan**

  Professor of Civil Engineering at Purdue University, his research interests are automated aerospace image mapping, spatial data modeling and analysis, remote sensing image processing and fusion and 3-D Geographic information systems and visualization, Dr. Shan holds a Ph.D in Photogrammetry and Remote Sensing from Wuhan Technical University of Surveying and Mapping, China, as well as Bsc. and Msc. From Zhengzhou Institute of Surveying and Mapping in China.

- **Professor Luk Van Wassenhove**

  Leading management thinker and educator. He is a professor at INSEAD, where he holds the Henry Ford Chaired Professorship in Manufacturing. He is also the Director of the Humanitarian Research Group and a Fellow of CEDEP, The European Center for Executive Education, based in France. His research and teaching are concerned with operational excellence, supply chain management, quality, continual improvement and learning. He currently leads INSEAD's Humanitarian Research Group.

Technical Advisory Committee Members

- **Mr. Stephen White**

  Career logistician with certifications in personal property and facilities management. Also trained in Six Sigma and process improvements. Currently Director of Property Management Division at Federal Emergency Management Agency. Holds an MBA in Industrial Management from Baker College Center for Graduate Studies and a BSc. from Limestone College

Role of the Technical Advisory Committee

Center for Infrastructure, Transportation, and the Environment

Role and Expectations

- The TAC is a requirement from USDOT, to ensure
  - The work addresses real-life needs
  - Quality of technical work
- Provide strategic guidance to the team
  - One-two electronic meetings
  - Or in any fashion you prefer…
  - Or at any time you chose…
- Serve a sounding board…
- Key principles, We:
  - Are appreciative of your participation…
  - Do not plan to overwhelm you…
  - Would like you to get involved as much as you desire…

The Problem

Center for Infrastructure, Transportation, and the Environment

Fact #1: Without access there is no response

- Fact #1: Disasters, snow storms, etc. frequently block access in transport networks
  - Without access, critical emergency functions such as search and rescue, distribution of relief supplies cannot start
  - Restoring access is critically important

2011年(平成23年)3月24日 木曜日 12版▲ 20

い物 欲しい人に

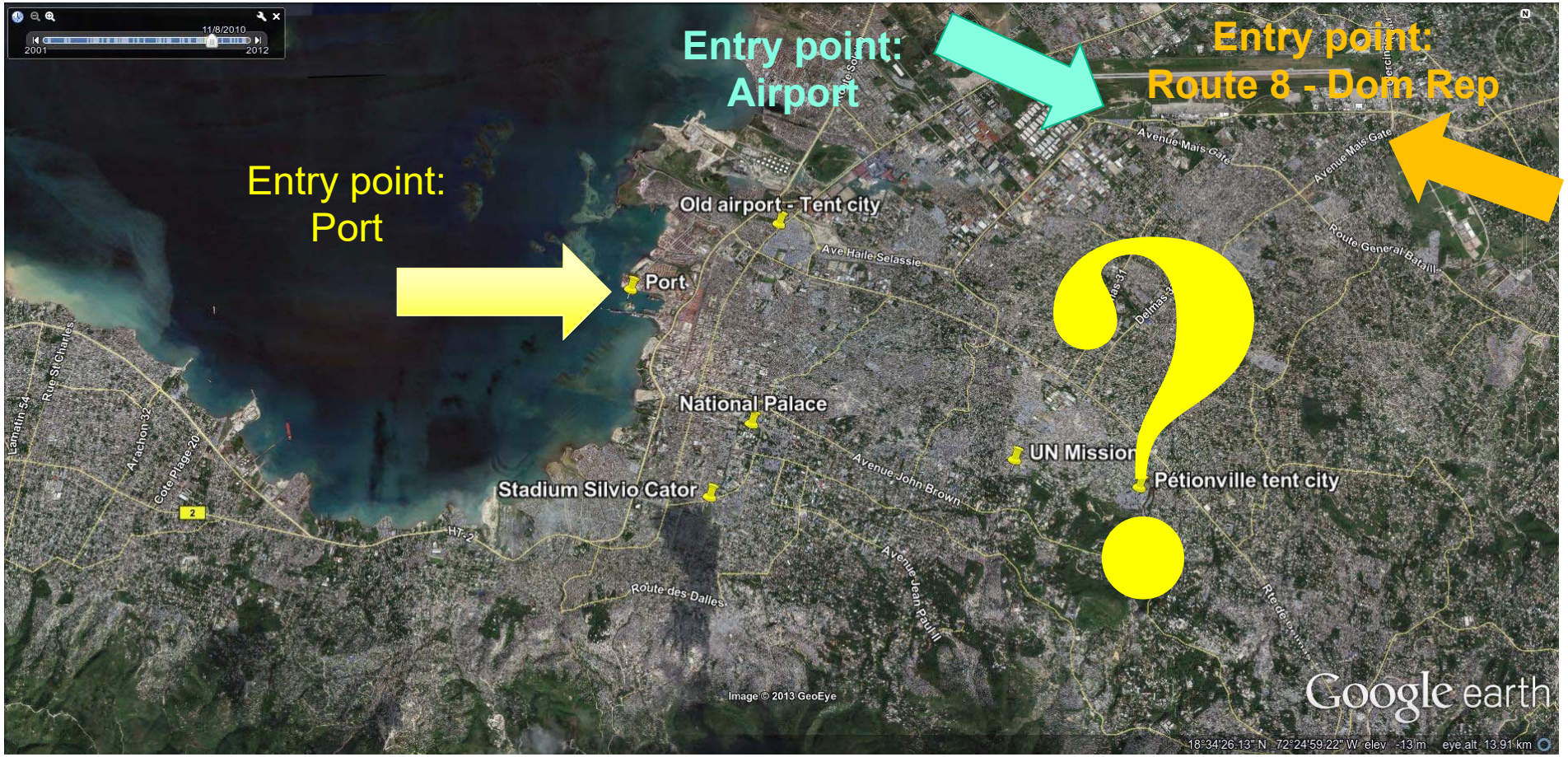
ニッポン みんなで

避難所「また水」「

Motoyoshi, Japan

Fact #2: Lack of visibility is a huge problem

- Responders typically arrive at the periphery, do not know what is happening inside the impacted area, cannot deploy their assets optimally

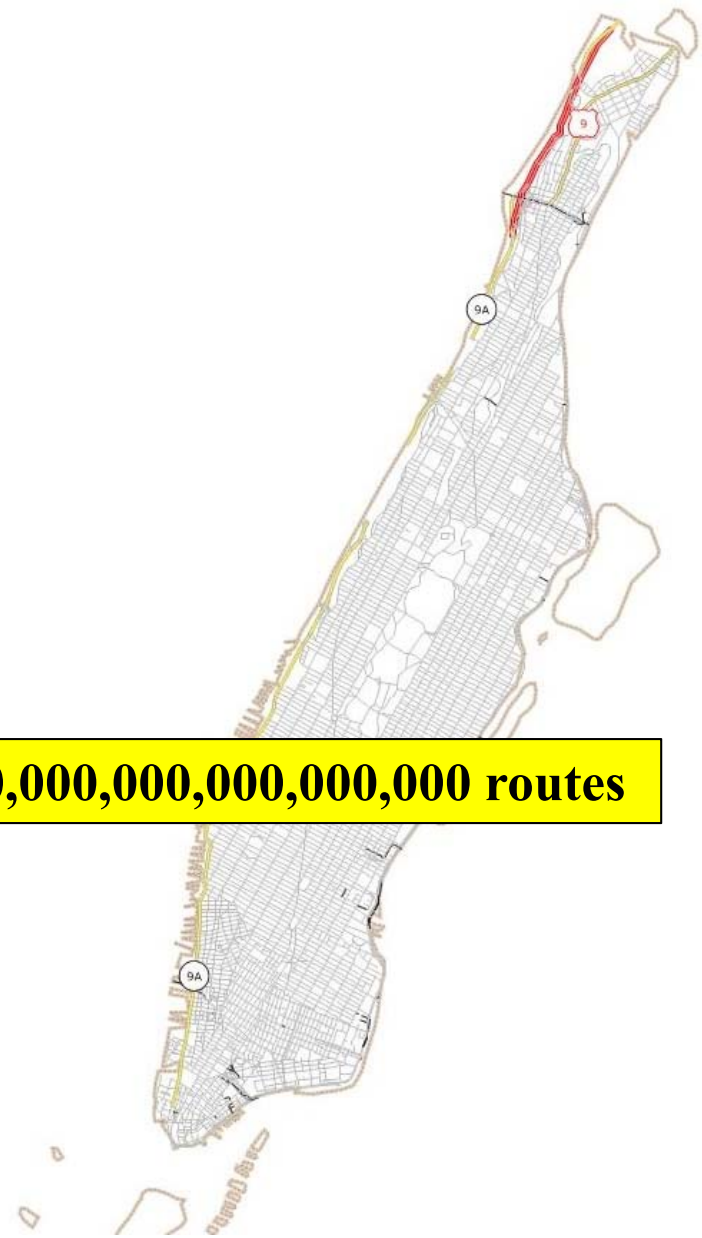


Fact #3: Deciding how to deploy is challenging

- One of the toughest math problems known to man…
- Imagine that after a snow storm 3’ of snow must be cleared to restore access in Manhattan
  - There are 5.15x1047 different routes to consider:

515,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000 routes

  - Finding the best route is beyond what any person could do, there are competing demands to be taken into account…



Fact #4: Coordination is a critical challenge

- Responder groups arrive to the site, frequently, guided by the news media's interpretation of the needs, leading to:
  - Overcrowding of resources and assets at some locations (even leading to fights…)
  - Lack of resources and assets at other locations
- More coordination could help
- Districting Plans that subdivide the work among responder groups could expedite access restoration

Center for Infrastructure, Transportation, and the Environment

Problems and Solutions…

- The Problems:
  - Fact #1: Without access there is no response
  - Fact #2: Lack of visibility is a huge problem
  - Fact #3: Deciding how to deploy is challenging
  - Fact #4: Coordination is a critical challenge

- The Solutions:
  - Increase the effectiveness of access restoration
  - Use RS technologies to "see" the impacts
  - Use state of the art computing to help decide
  - Develop Districting Plans to subdivide the work

Center for Infrastructure, Transportation, and the Environment

Vision and Goals

Center for Infrastructure, Transportation, and the Environment

Vision

- A state-of-the-art decision support system (DSS) that:
  - Uses commercial remote sensing (CRS) assessments of network conditions and disaster impacts to produce an optimal access restoration plan (ARP)
  - Helps responders optimally use their scarce resources to:
    - Orchestrate the opening of roads
    - Prioritize work
    - Facilitate the cooperation of multiple groups
- The DSS will use multi-modal temporal data to update estimates of disaster impacts, and use optimization to refresh the ARP as new data are available

Center for Infrastructure, Transportation, and the Environment

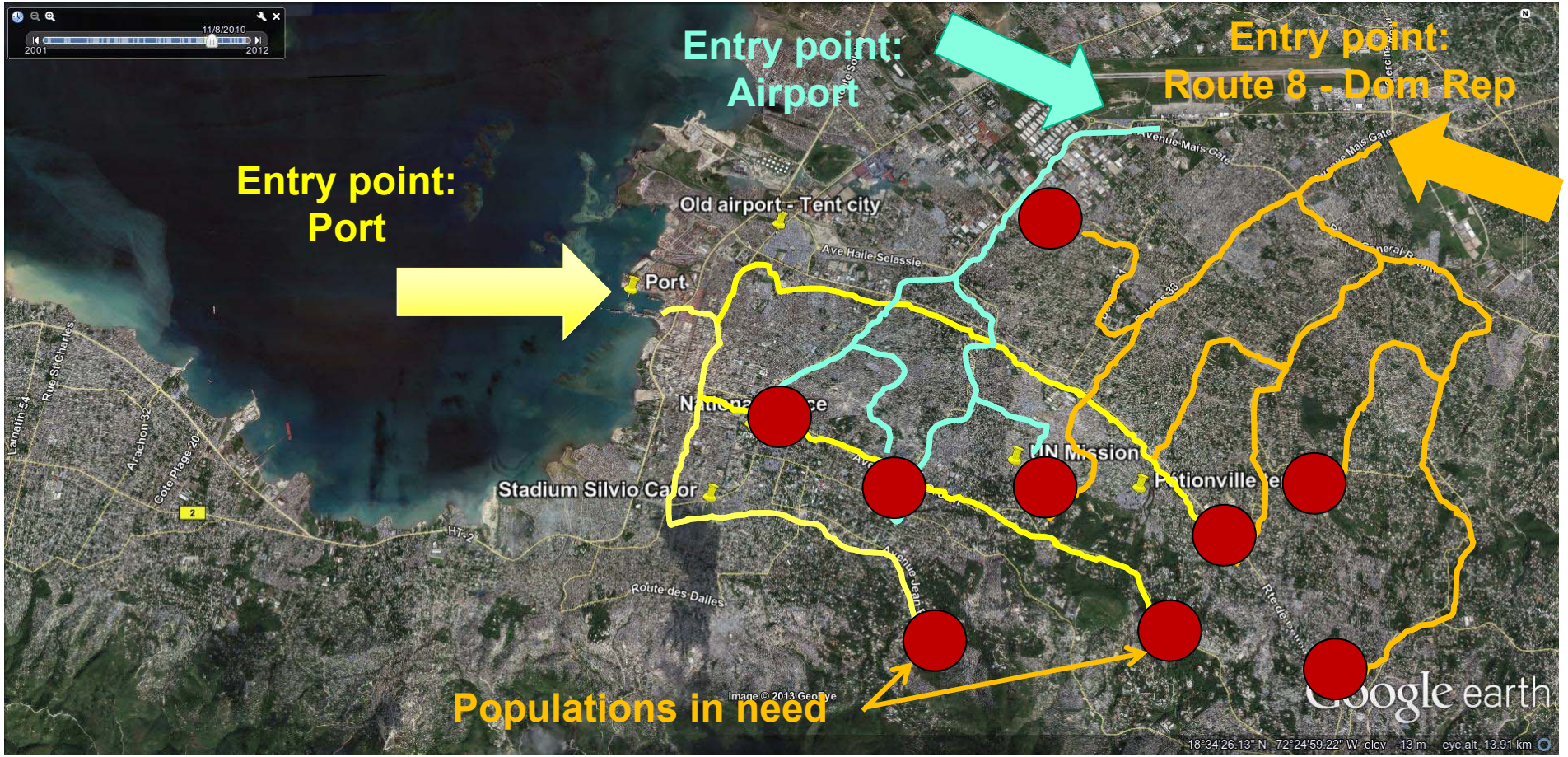
Goal and Project Objectives

- Long term goal:
  - To assess the conditions of transportation networks and the overall impacts on the built environment and population
  - To determine the social benefits associated with the access restoration, and the amount of time and effort required to restore the various links to operational condition
- Project objectives:
  - To develop techniques to assess social benefits, and determine optimal access restoration considering:
    - Debris obstructing roads
    - Flood impacts

Center for Infrastructure, Transportation, and the Environment

Access Restoration Plan

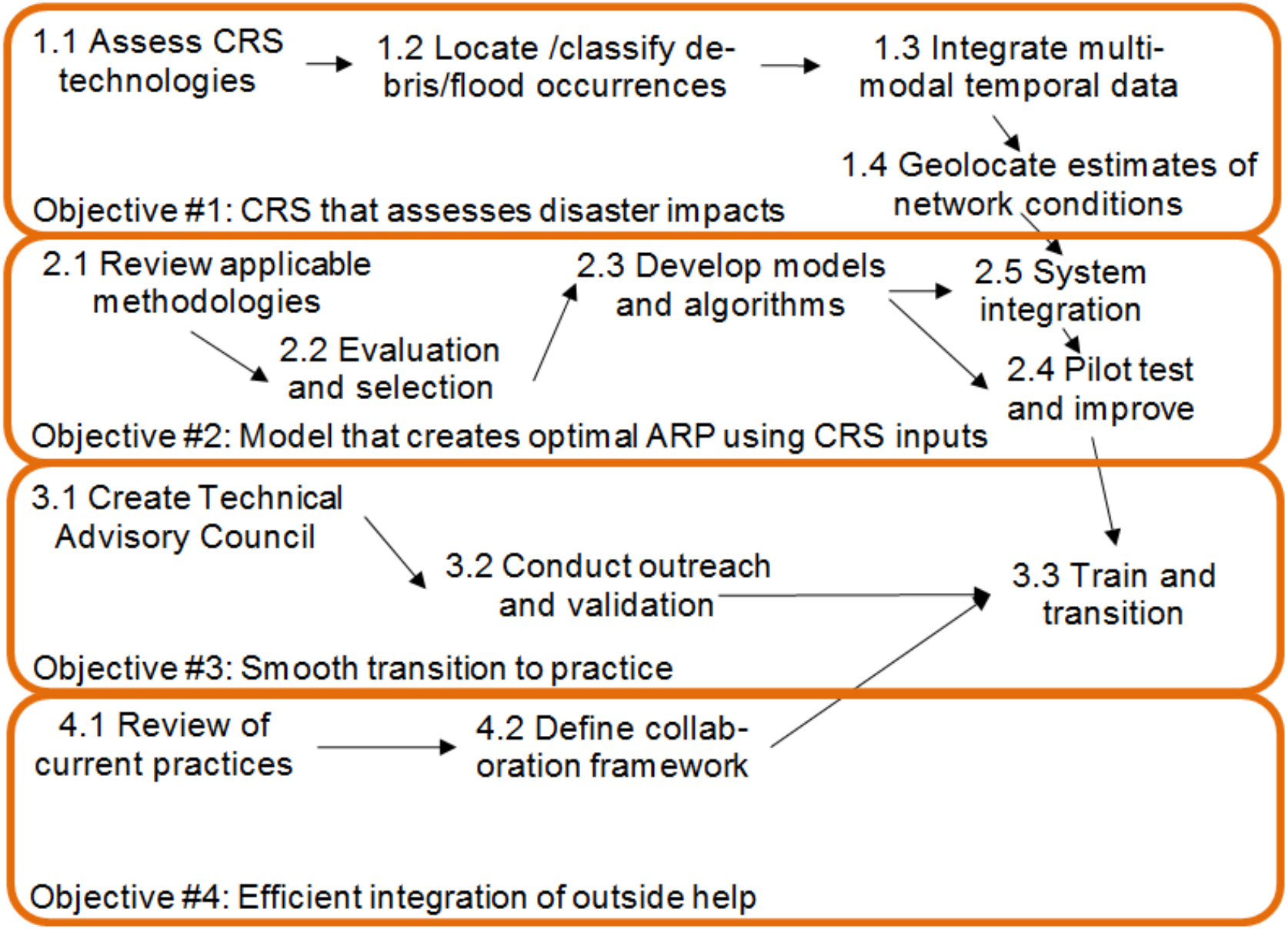
- The Access Restoration Plan will outline in what order to open roads from the various entry points available in an "optimal" fashion, responding to needs



Project Overview

Center for Infrastructure, Transportation, and the Environment

Key Objectives and Tasks



Development of CRS Techniques

Jan van Aardt,

Associate Professor,
Acting Director, Digital Imaging and
Remote Sensing Group, Chester F. Carlson
Center for Imaging Science,
Rochester Institute of Technology
vanaardt@cis.rit.edu

1.1 Assess CRS Technologies

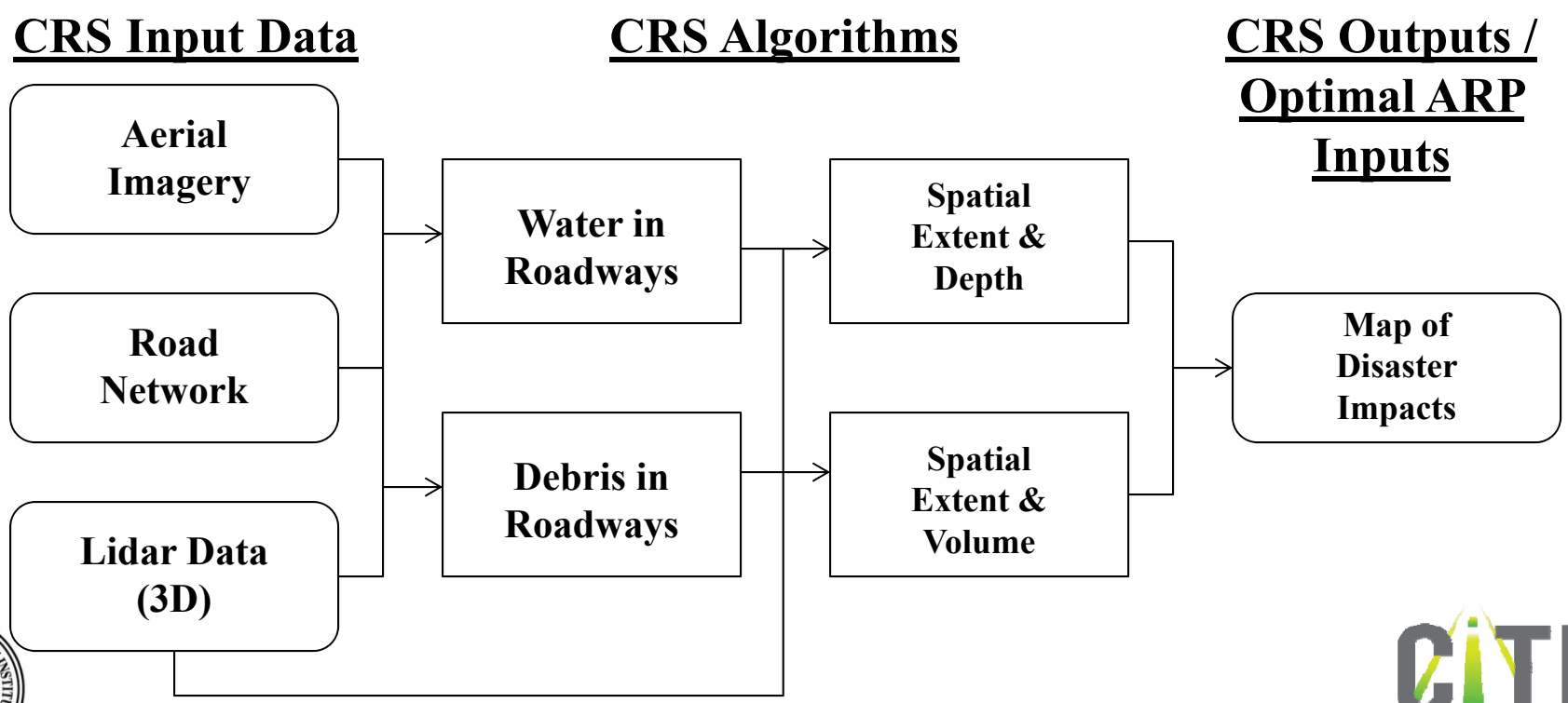
- Prepared Remote Sensing Guidance Document for NYCDOT
- Analysis of various remote sensing modalities
  - Remote sensing technical principles
  - Visible (color), infrared, multispectral, hyperspectral, lidar, synthetic aperture radar
  - Capabilities and limitations
- Remote sensing platforms/sources (aircraft, satellites)
- Disaster scenarios, Essential Elements of Information (EEIs)
  - Flood, Earthquake, High Wind, Man-made (explosive)
  - EEIs in the context of transportation infrastructure
- Mapping of modalities to EEIs and scenarios via Decision Support Matrices
- Suggest follow-on discussions with NYC DOT to refine to more NYC-specific EEIs

1.1 Assess CRS Technologies

					Daylight Only			Day / Night	
Scenario	Characteristics	Essential Elements of Information (EEIs)	Spatial Resolution / Point Density	Geolocation Accuracy	Color	MSI/HSI	SWIR	MWIR / LWIR	LiDAR
Flood	*Inundated roads and facilities. Debris deposits on roads. Washouts, undermined roads. Damage to bridges and bridge approaches. Hazardous effluents from inundated storage or transmission facilities.*	Location of current inundation boundaries	0.5m	2.0m	X	X	X		X
		Location of high water boundaries	0.1m vert, 1.0 m horiz	2.0m	X	X			
		Debris location and passability	0.2m	1.0m	X				X
		Washout / undermining of roads location and severity	0.1m	1.0m	X				X
		Presence of effluents in flood waters	0.5m	2.0m	X	X		X	
		Presence of effluent residue	0.5m		X	X			
		Identification / quantification of effluents	1.0m	2.0m		X			
		Bridge damage assessment	0.1m		X				X

1.2 Locate and Quantify Flood/Debris

- To assess disaster impacts on the transportation network, flooding and build up of debris in the road network are automatically detected and characterized using CRS data collected after the disaster



Center for Infrastructure, Transportation, and the Environment

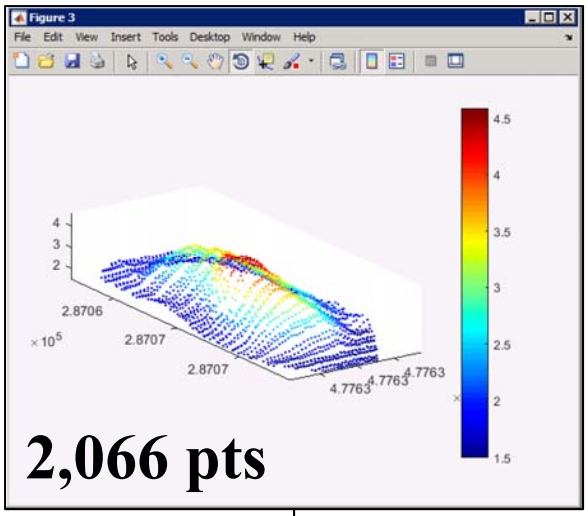
Debris Detection & Volume Estimation Algorithm

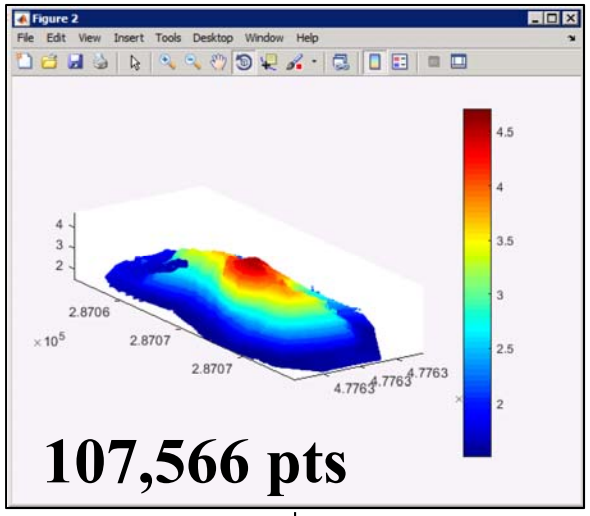
- The goal of this algorithm is to take lidar points from the road network, detect piles of debris, and estimate their volumes. At a high level:
- 1) Statistical outlier removal
- 2) Surface normal & normal distribution estimation
- 3) Vegetation removal
- 4) Segment into individual objects using region growing
- 5) Discard "non-debris" regions based on properties such as height variation, normal distribution, etc.
- 6) (Optional) Discard additional false alarms using object texture properties obtained from imagery
- 7) Model surfaces with alpha shapes to obtain volume

Center for Infrastructure, Transportation, and the Environment

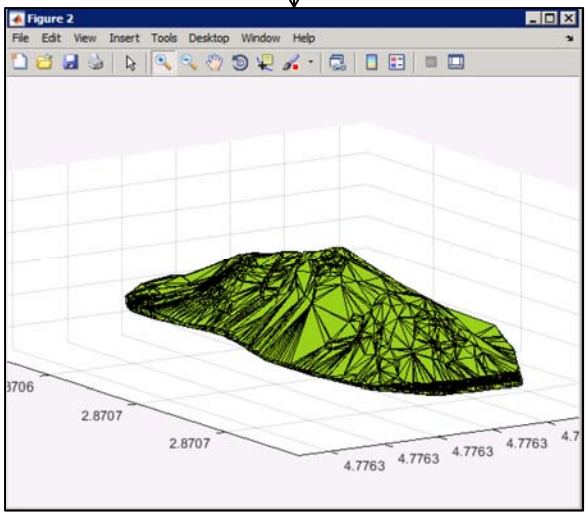
Validation of Volume Estimation Capabilities

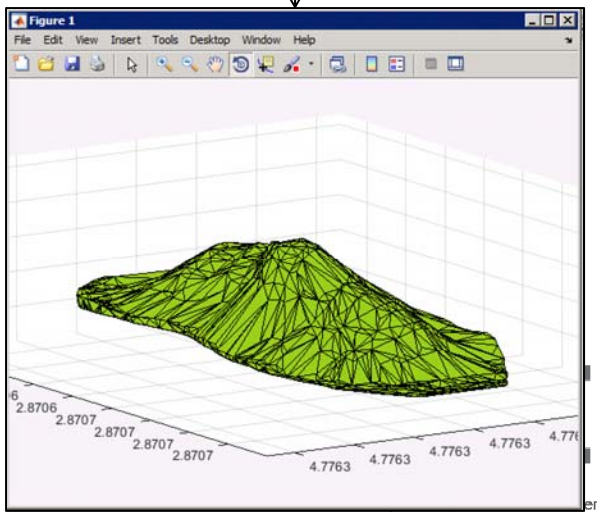
- Extremely high resolution terrestrial lidar scans were used to obtain a "ground truth" volume and then volumes were estimated based off of moderate resolution (20-25 pts/m^2) airborne scans. Alpha shapes were used to model the debris surfaces and obtain volumes.





Alpha Shapes

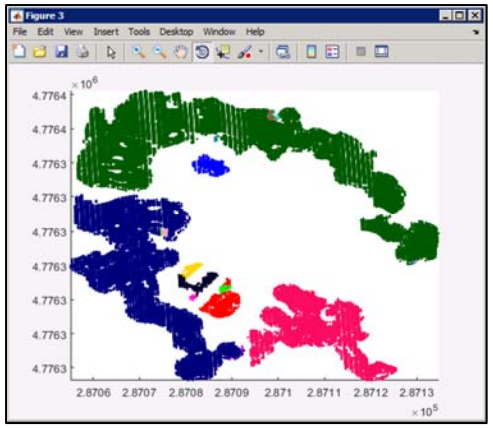




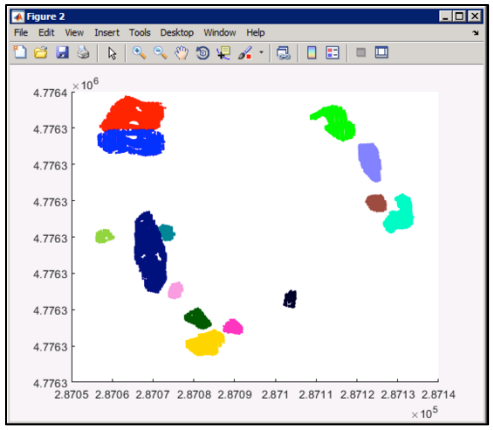
The study showed that at this resolution, volume estimation using airborne scans matched ground truth with 94.7% accuracy.

Debris Detection on Validation Debris Piles: Site1

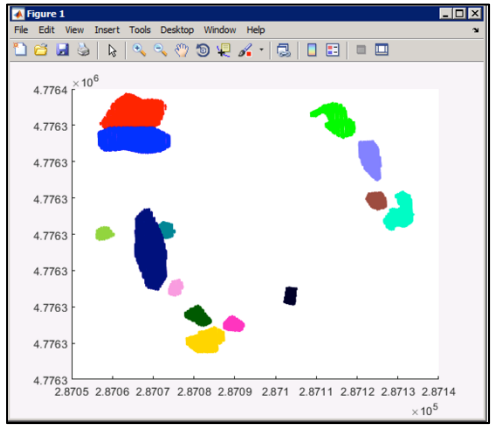
- Debris detection & volume estimation applied to entire point cloud of each debris site. Debris piles found within boundaries of ground truth piles extracted for volume comparison.



All Detected Piles

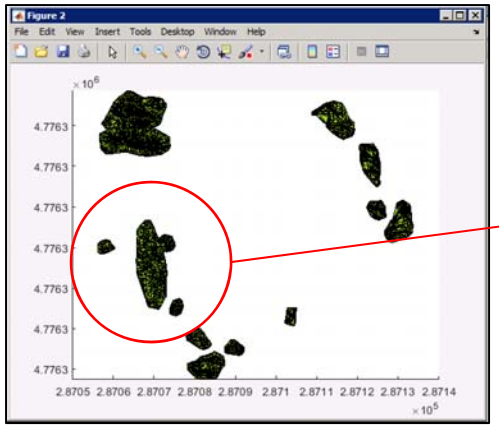


Detected Validation Piles

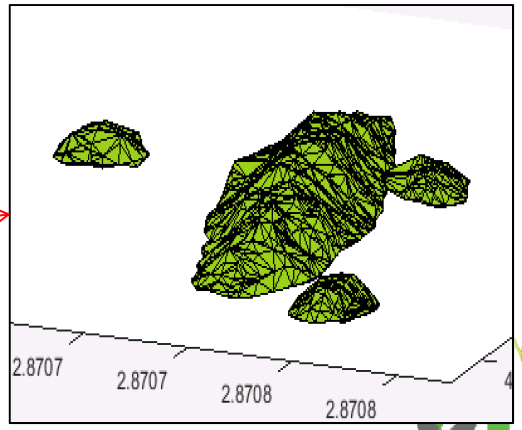


Ground Truth Validation Piles

14 of 14 debris piles were detected with a mean volume error of 2.67%



Modeled Debris Piles

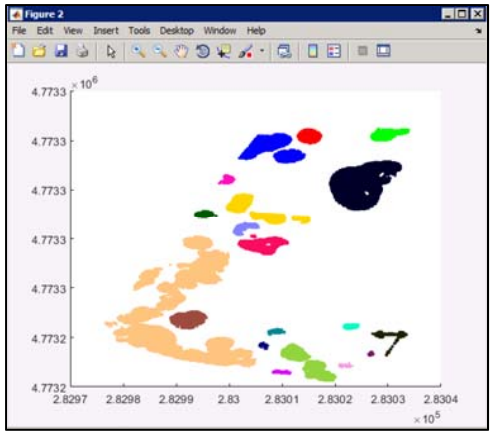


Zoom View

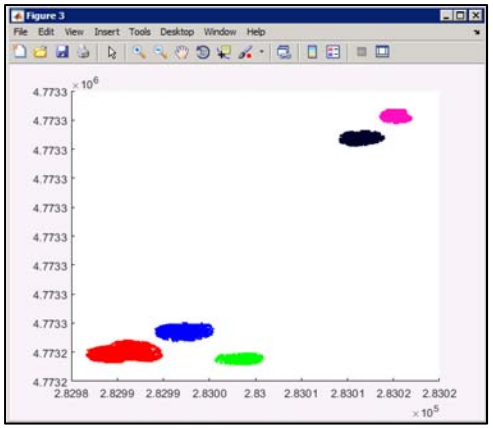
Center for Infrastructure, Transportation, and the Environment

Debris Detection on Validation Debris Piles: Site2

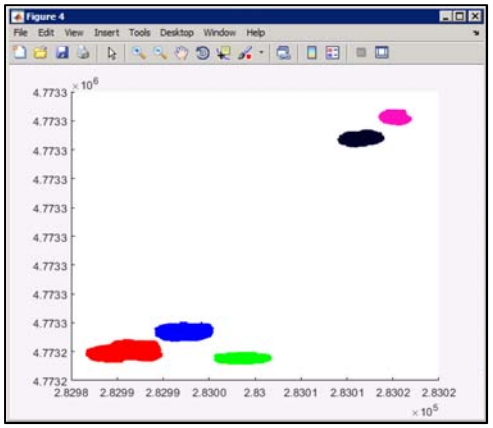
- Debris detection & volume estimation applied to entire point cloud of each debris site. Debris piles found within boundaries of ground truth piles extracted for volume comparison.



All Detected Piles

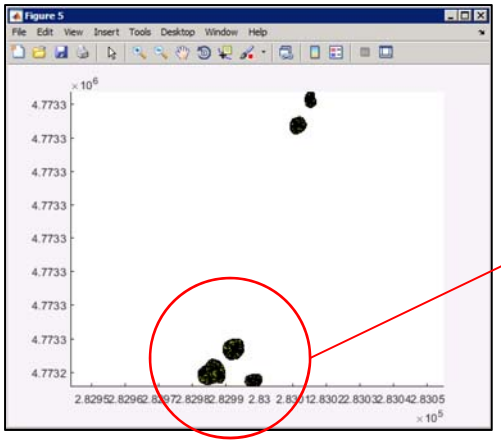


Detected Validation Piles

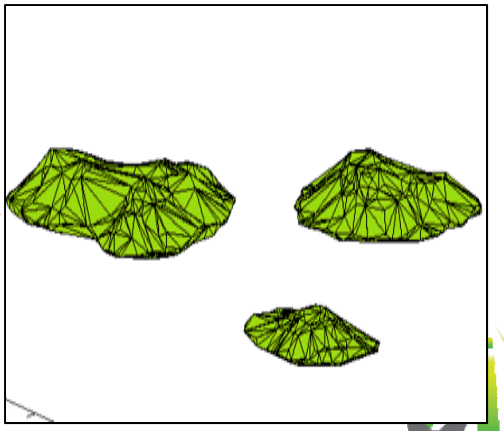


Ground Truth Validation Piles

5 of 5 debris piles were detected with a mean volume error of 5.19%



Modeled Debris Piles

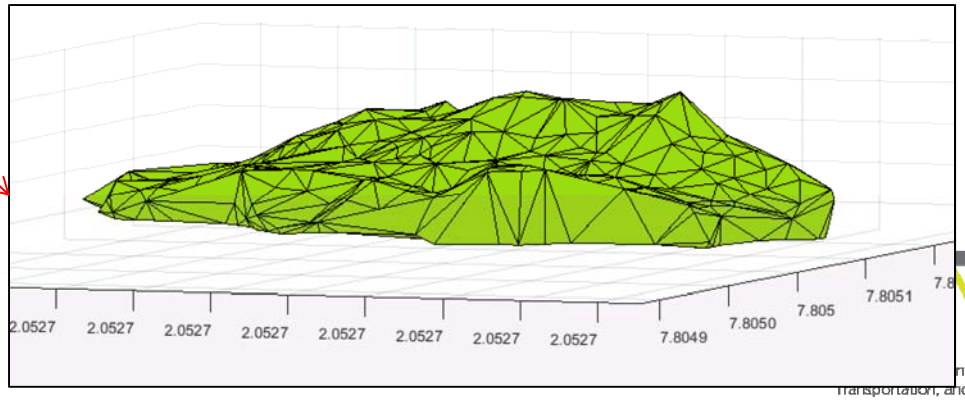


Zoom View

Center for Infrastructure, Transportation, and the Environment

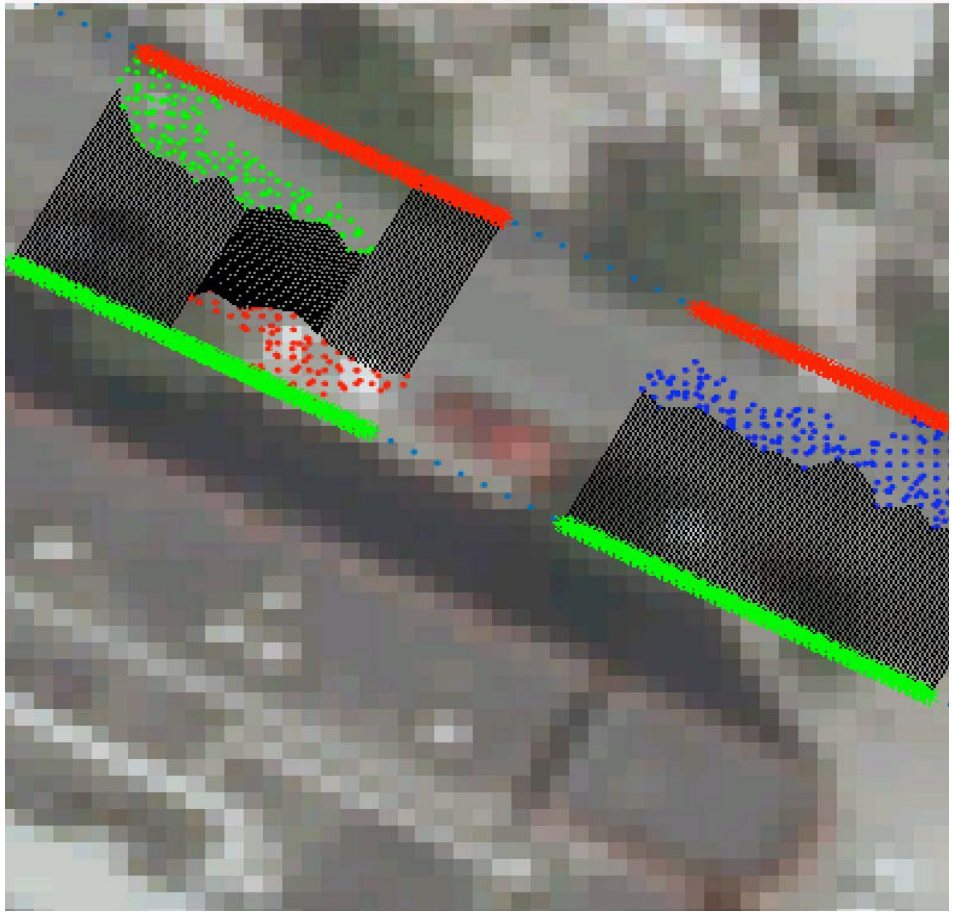
Debris Detection on Real Disaster Data

- Lidar data from the Haiti earthquake were used to test the debris detection algorithm. In this case the only means of validation is visual inspection.



Road Metrics

- For each road with debris, several metrics are computed:
  - Passable width
    - How much room is there for a vehicle to pass through?
  - Penetration depth of debris
    - How far from the road edge does the debris extend?
  - Incremental volume of debris
    - How much debris do I have to remove to clear another 1ft of width? 2ft? Etc.



- All of this information is distilled into a "debris map" shapefile

Center for Infrastructure, Transportation, and the Environment

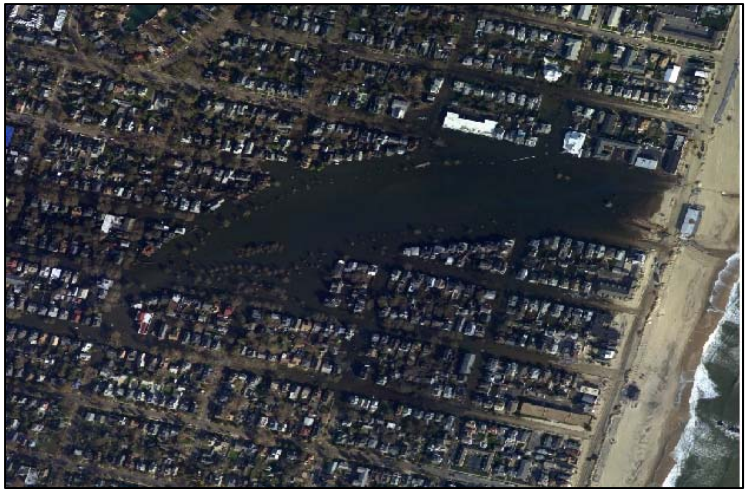
Flood Detection and Depth Estimation Algorithm

- The goal of this algorithm is to find flooded areas in the roadways and estimate the depth of the water. This algorithm has not yet been developed, but will be based off refinements to previous work done at RIT. At a high level:
- 1) Find a pixel that contains water to use as a seed
- 2) Perform target detection using the seed pixel as a signature
- 3) Determine the elevation at the edge of the water
- 4) Compute water depth per pixel using an existing DEM and the elevation difference from the edge

- This will be tested using color imagery, multispectral imagery, and lidar.

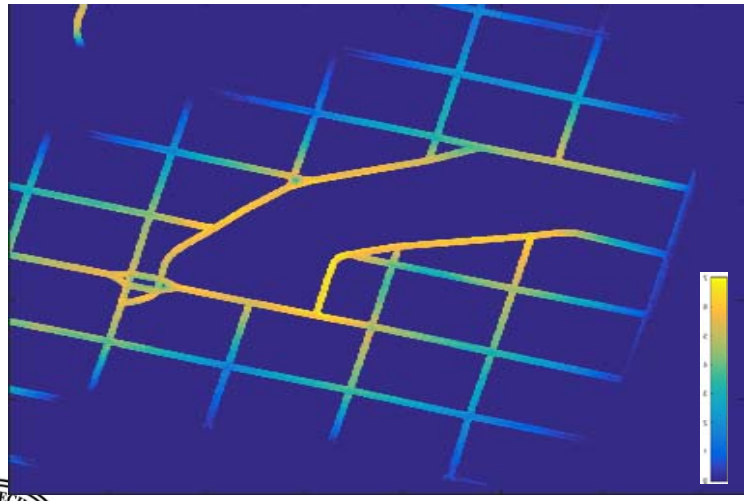
Center for Infrastructure, Transportation, and the Environment

Locate Flooding and Estimate Depth

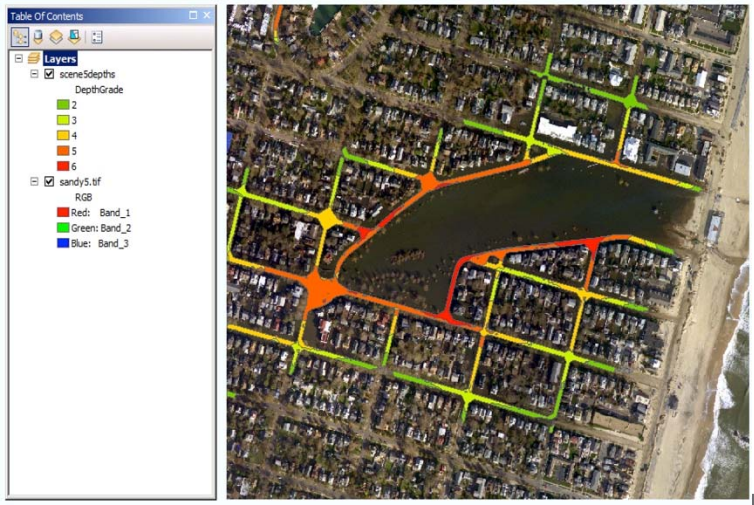


Aerial Image

Detected Water in Roadways



Depth Computed from Lidar



Depth Map

Center for Infrastructure, Transportation, and the Environment

Mathematical Modeling

José Holguín-Veras,

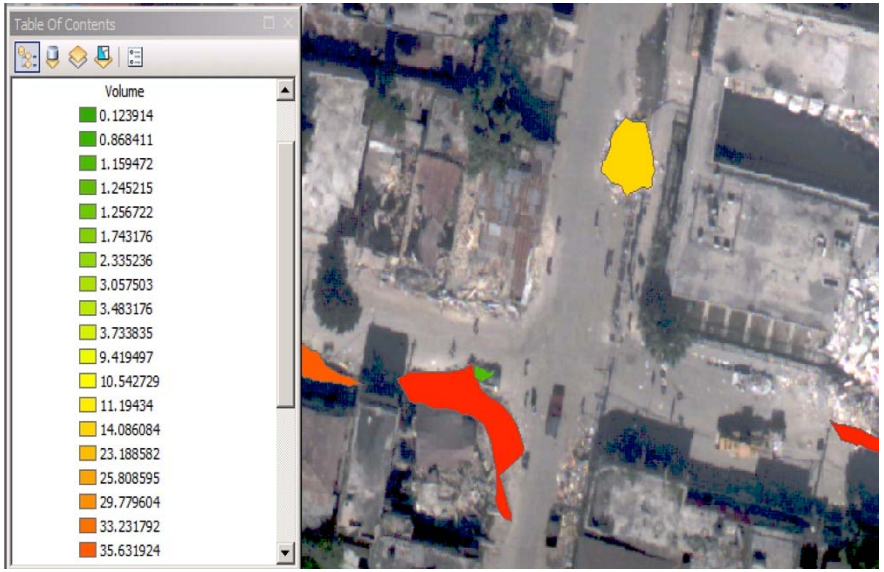
William H. Hart Professor,
Director of the Center for Infrastructure,
Transportation, and the Environment
Rensselaer Polytechnic Institute
jhv@rpi.edu

Main objectives

- To develop mathematical procedures that obtain the "Optimal Access Restoration Plan" that account for:
  - The time/effort required to restore access to the links that comprise the transportation network (using CRS estimates)
  - The impacts that the disaster produced on the population
  - The impacts that delays in restoring access would have on the population and the economy
- These elements must be captured in appropriate objective functions and metrics that
  - Combine these elements, accounting for their importance
  - Lead to the best access restoration plans possible

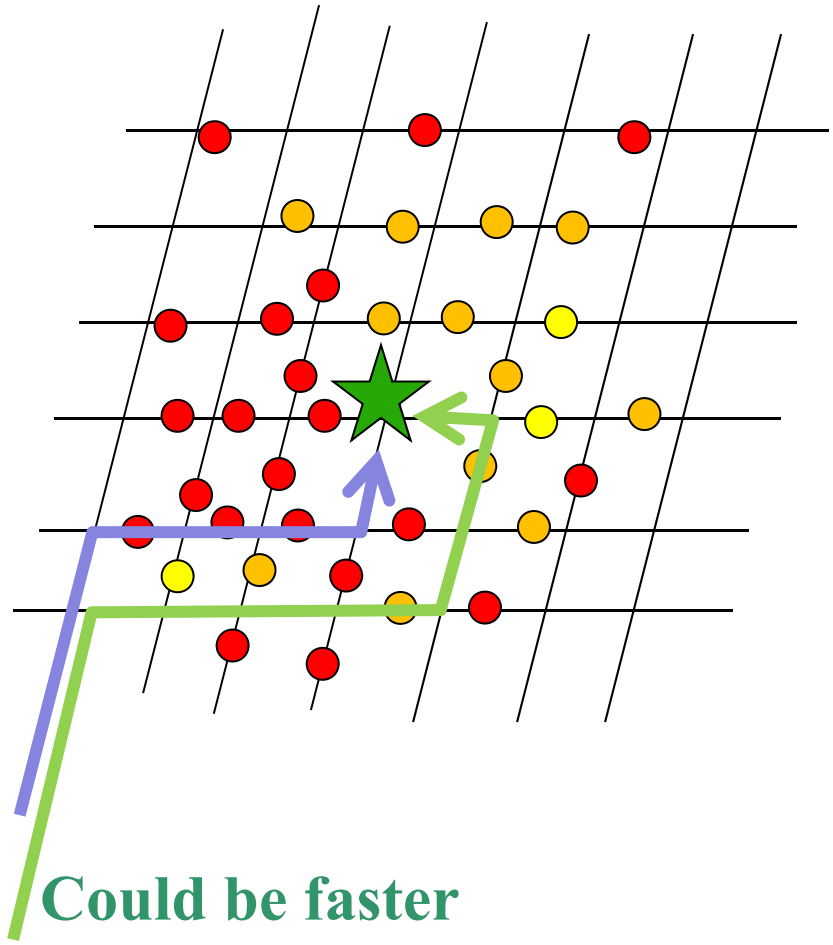
Center for Infrastructure, Transportation, and the Environment

Estimation of time/effort required to open links



Specific links

A transportation network



Could take a lot of effort and time

Could be faster

RPI is designing the algorithms to perform these computations, estimating the impacts of restoration time on population welfare

2.1 Review of Metrics

- Many different metrics possible (and used in practice)
  - It is important to assess how good they are
  - The correct metric must consider human suffering and the cost of restoring access
- Model formulations should be based on welfare economics: using microeconomic techniques to evaluate well-being at the aggregate level
- Two primary components:
  - Private costs (restoration costs, restoration time)
  - External costs (deprivation costs, impacts of restoration delays on population and the economy)

2.1 Review of Metrics

Model Class	Objective Functions
Social cost model: private and deprivation costs, all nodes and time epochs	$\Psi_T^{SC}(X,T)=\Omega_T(X,T)+\sum_{j=1}^{J}\Gamma_{i_j}(X,t_j)+\sum_{i=1}^{N}\Gamma_i(X,T)$
Variable penalty model: logistic costs and variable penalties, all nodes and time epochs	$\Psi_T^{VP}(X,T)=\Omega_T(X,T)+\sum_{i=1}^{N}\sum_{j=1}^{J}\rho'\left(\delta'_{it_j}-\delta_p\right)y'_{it_j}$
Constant penalty model: logistic costs and penalties, all nodes and time epochs	$\Psi_T^{CP}(X,T)=\Omega_T(X,T)+\sum_{i=1}^{N}\sum_{j=1}^{J}\rho y_{it_j}$
Hard constraint model: Constrained summation of logistic costs	$\Psi_T^{HC}(X,T)=\Omega_T(X,T):\delta_{it_j}\leq\delta_p$
Unmet demand model: Minimize total unsatisfied demand	$U^*=\sum_{i=1}^{N}\sum_{j=1}^{J}U_{it_j}$

2.2 Evaluation and Selection of Priority Metrics

- Priority metrics provide the objective of the restoration of access. This will help decide which roads need to be opened first and/or which points in the disaster zone have the highest priority
  - Classification of roads: primary, secondary and tertiary
  - Maximum Population
  - Minimum Social Cost
  - Minimum Marginal Cost
  - Minimum Social Cost Divided by Population
  - Minimum Marginal Cost Divided by Population

2.3 Mathematical Model/Algorithms

$$MIN \quad SC = \sum_{d \in D} \pi_d \, \gamma(\delta_d) + \Omega(X,T) \quad (1)$$

Social costs minimization

Population Impacts

Restoration Costs

$$\sum_{(i,j)\in A} x_{ij}^{d\tau_{ij}} \geq 1 \qquad i \in S, d \in D \qquad (2)$$

$$\sum_{(i,d)\in A} \sum_{t\in T} x_{id}^{dt} \geq 1 \qquad d \in D \qquad (3)$$

$$\sum_{(i,k)\in A} x_{ik}^{dt} - \sum_{(k,j)\in A} \sum_{\hat{t}=t}^{T} x_{kj}^{d\hat{t}} \leq 0 \qquad d \in D, k \in N \setminus (S \cup d), t = 1 \ldots T \qquad (4)$$

Network design

$$\alpha_{ij}^{t} - x_{ij}^{dt} \geq 0 \qquad (i,j) \in A, d \in D, t = 1 \ldots T \qquad (5)$$

Scheduling

$$\sum_{(i,j)\in A} \sum_{s=t}^{t+\tau_{ij}-1} \alpha_{ij}^{s} \leq q_t \qquad t = 1 \ldots T \qquad (6)$$

Capacity

$$\delta_d \geq \sum_{t\in T} \sum_{(i,d)\in A} t x_{id}^{dt} \qquad d \in D \qquad (7)$$

Deprivation time

$$x_{ij}^{dt}, \alpha_{ij}^{t} \in \{0,1\}, \delta_d \geq 0 \qquad d \in D \qquad (7)$$

Integrality

2.3 Mathematical Model/Algorithms

- Determines the sequence of roads (links) to be restored to minimize human suffering (dep costs)
  - Deprivation costs
  - Scheduling the restoration of roads
  - Capacity constraints
- Allows analysis of the impact of capacity constraints in the entire planning horizon
- A number of algorithms have been developed and tested…

Center for Infrastructure, Transportation, and the Environment

Path Access Restoration (PathAR) Heuristic

- Step 1: begin with a sub-tree T of the graph G consisting of only the super node z1, k=1
- Step 2: if k=n go to Step 4, otherwise go to Step 3
- Step 3: determine a terminal $z_{k+1} \notin G$ according to priority metric. Add to T the shortest path joining it with z_{k+1}. k=k+1. Go to Step 2
- Step 4: determine a MST for the sub-network induced by the vertices in T
- Step 5: delete from the MST in non-terminals of degree 1. Stop

Center for Infrastructure, Transportation, and the Environment

Project Status

Overview

- Things are going reasonably well
- Complex undertakings like this one face challenges:
  - Delays getting data:
    - November 20/2014: Kick-off meeting, data requested
    - Jan 20/2015: Second request for imaging data
    - March 30/2015: Third meeting to discuss request
    - May 22/2015: Meeting with NYCDOT NYC ITT to brief
    - July 20/2015: 30GB of data received
    - September 4/2015: Update from NYCDOT, still working on legal paperwork to obtain LIDAR data
  - Personnel changes:
    - M. Jaller and F. Aros got faculty positions
    - Replacing them has taken time…

Center for Infrastructure, Transportation, and the Environment

Project Progress

Phases	Tasks	Progress
Development of CRS Techniques		43%
Mathematical Modeling		37%
Validation and Dissemination		46%
Integration Procedures		20%
TOTAL		**40%**

Concluded In Progress Forthcoming

Center for Infrastructure, Transportation, and the Environment

Next Steps

- Task 1.3: Integrate multi-modal non-CRS
  - Development of DSS Specifications and Systems for the collection, analysis, and visualization of remote sensing data, including a wide sensor array from transportation sensors such as CCTVs, Traffic Monitoring Stations (TMS), traffic signals, and loop detectors.

- Task 1.4: Geolocate Impacts
  - The location and quantification information will be generated in geographic information system (GIS) format, which will enable integration with mathematical models.

Center for Infrastructure, Transportation, and the Environment

Next Steps

- Task 4.1: Review current practices.
  - Conduct a comprehensive review and analysis of current integration practices
- Task 4.2: Definition of the Integration Framework.
  - Building on the review, the team will define the characteristics needed to develop an integration and coordination framework that considers pre-established stakeholders, while efficiently integrating outside help.
  - The framework will then integrate with the proposed DSS.

Center for Infrastructure, Transportation, and the Environment

Discussion

Center for Infrastructure, Transportation, and the Environment

Thanks!

Center for Infrastructure, Transportation, and the Environment

APPENDIX E:
CRS Playbook

Remote Sensing
Disaster Response Playbook
for
NYC Department of Transportation

Decision Support System for Optimal Access Restoration in Post Disaster Environments

Jan van Aardt (RIT / Co-PI)

José Holguín-Veras (RPI / PI)

July 2015

Introduction

- Objectives
- Key remote sensing physics and technical concepts
- Remote sensing data sources
- RIT remote sensing system
- Remote sensing data visualization tools

Why Remote Sensing for Disaster Management?

- Not the answer to every problem, but for large scale incidents can provide:
  - Rapid, synoptic coverage of large areas
  - Detection of hazardous materials
  - Damage assessment
  - Flood mapping
  - Overall situational awareness

Remote Sensing is More Than Color Ortho Imagery

Traditional color ortho-imagery is very useful but:

- Other "modalities" are important as well such as:
  - Light Detection and Ranging (LiDAR)
  - Thermal Infrared (TIR): Longwave Infrared (LWIR) & Midwave Infrared (MWIR)
  - Multispectral Imaging (MSI)
  - Hyperspectral Imaging (HSI)

- The end product is most important
  - decision support <u>information</u>
  - RS data needs to be seen as a element of information to support decision making

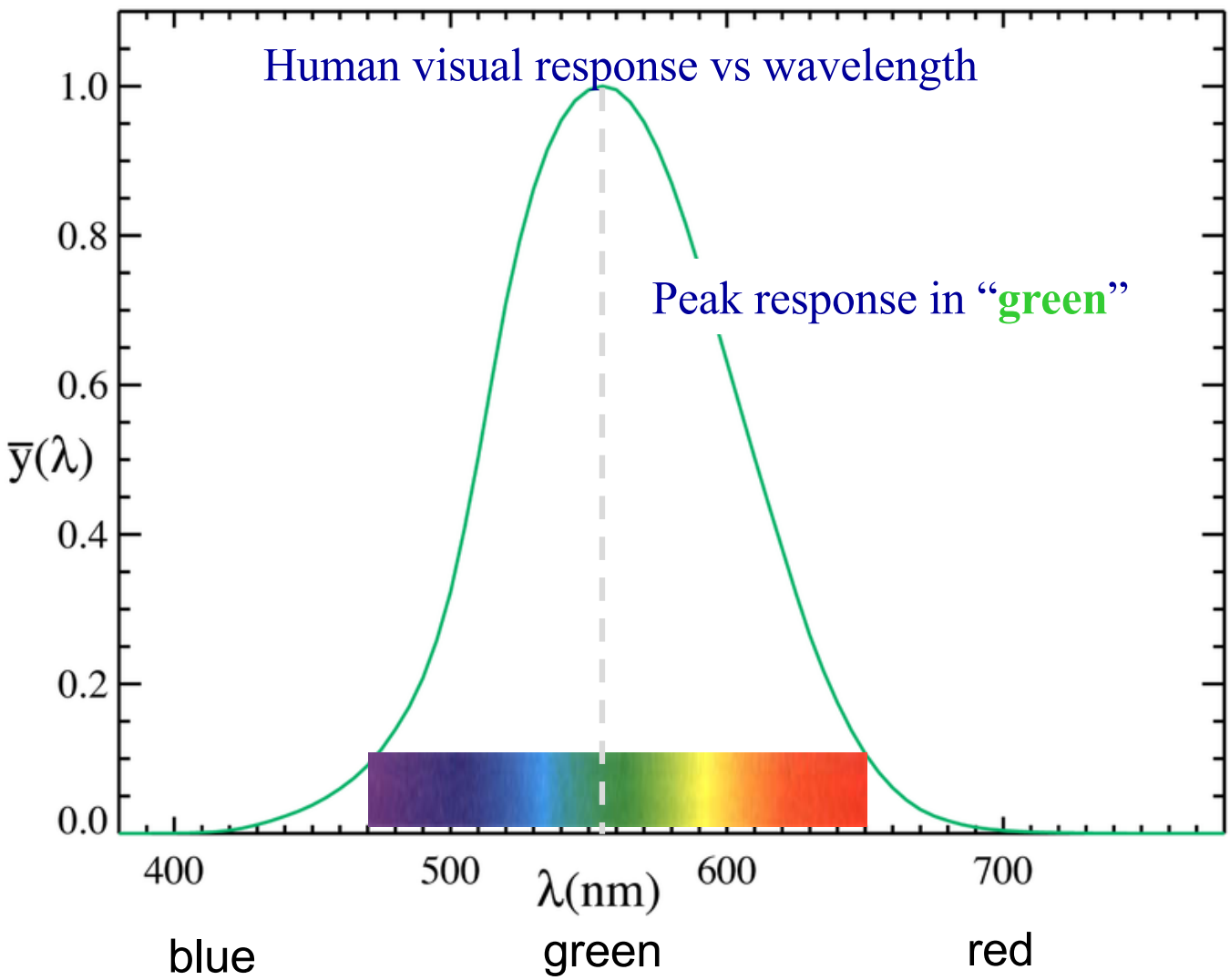
Purpose of the Playbook

- Users need to understand what technology can do
  - What opportunities are there to help based on current capabilities?
  - Realistic expectations about what is possible and what is not

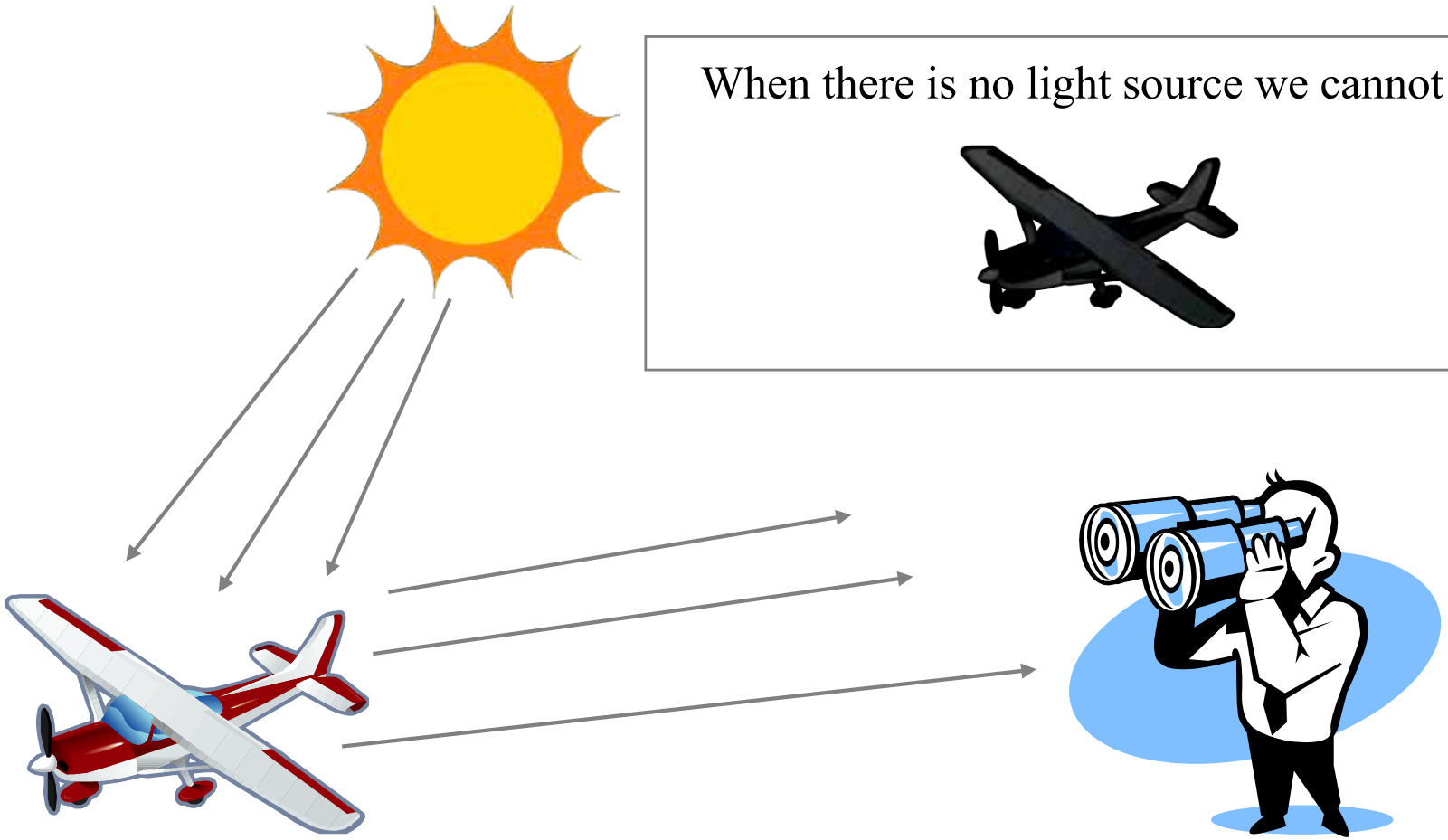
- Provide guidance for
  - Selection of appropriate modalities and data products
  - Tasking
  - Cost and schedule implications
  - Operational issues
  - Vendors

NATURE OF LIGHT

Human Visible Wavelengths

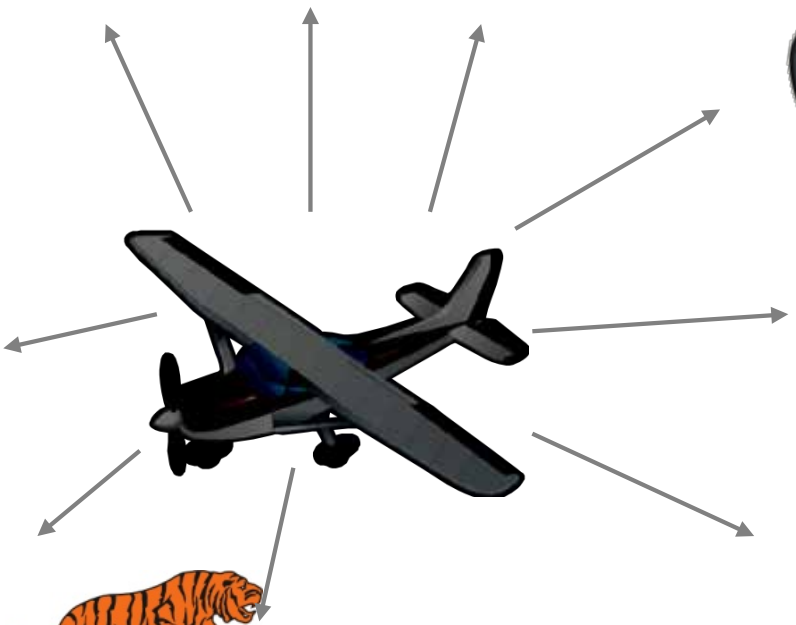


Our Eyes "See" Reflected Energy From the Sun



Objects Emit Energy According To Their Temperature

- We can't see this with our eyes, but it is there
- Special materials and cameras can detect this energy in the infrared spectrum



Infrared camera

This works day or night since it uses the object's own energy as the "light source"

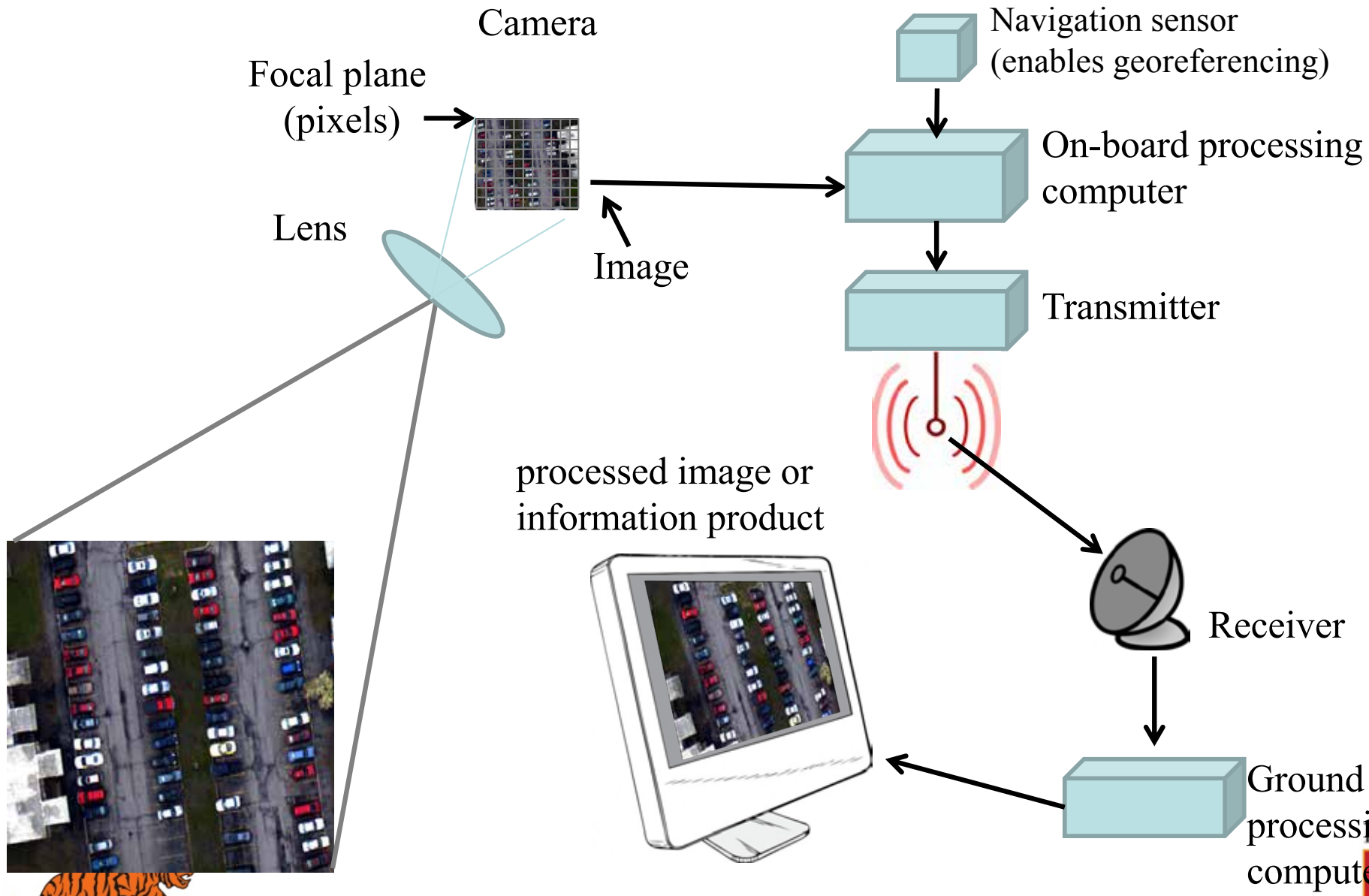
Typical Temperature and Wavelength Regions

As temperatures get cooler, radiated energy moves further into the infrared

• Room Temperature	20 C	Infrared	"Invisible"
• People	37 C		
• Charcoal	400 C		
• Hot fire	1,000 C	Orange-Red	
• Light bulb	2,500 C	Yellow	"Visible"
• Sun	5,500 C	Blue-green	

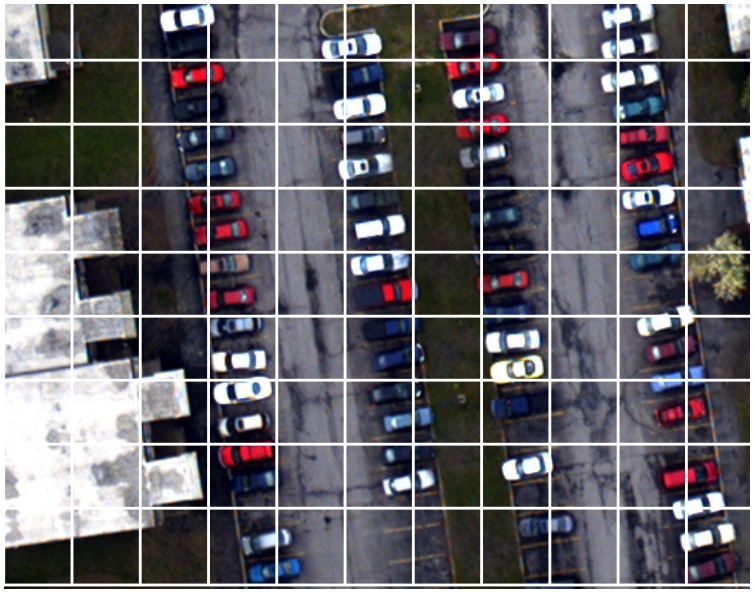
IMAGING SYSTEM BASICS

Basic Remote Sensing System Architecture

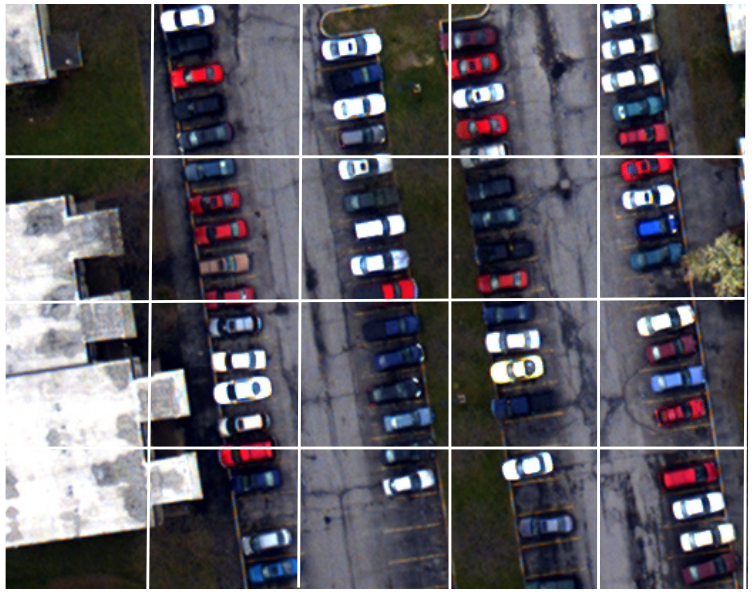


Spatial Resolution

“Spatial resolution” is a measure of how many pixels are on a target



“high” spatial resolution has many pixels on target

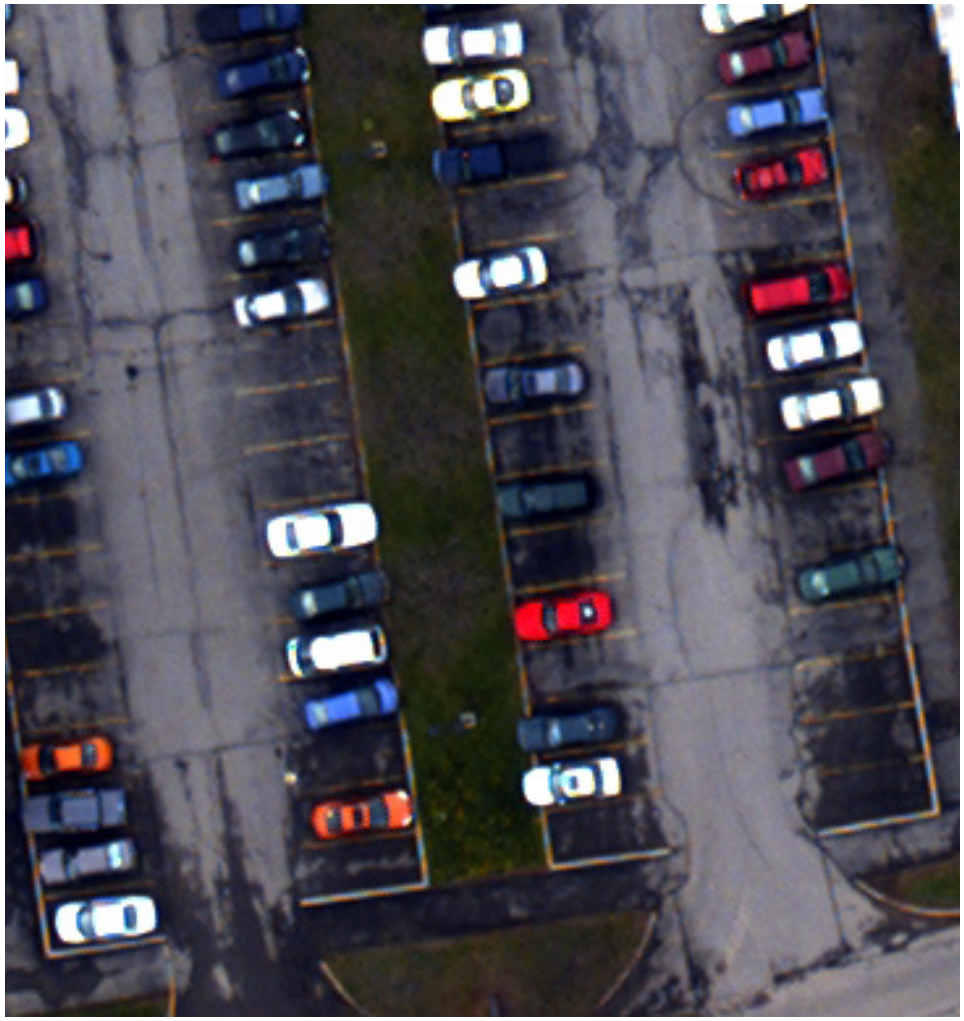


“low” spatial resolution has few pixels on target

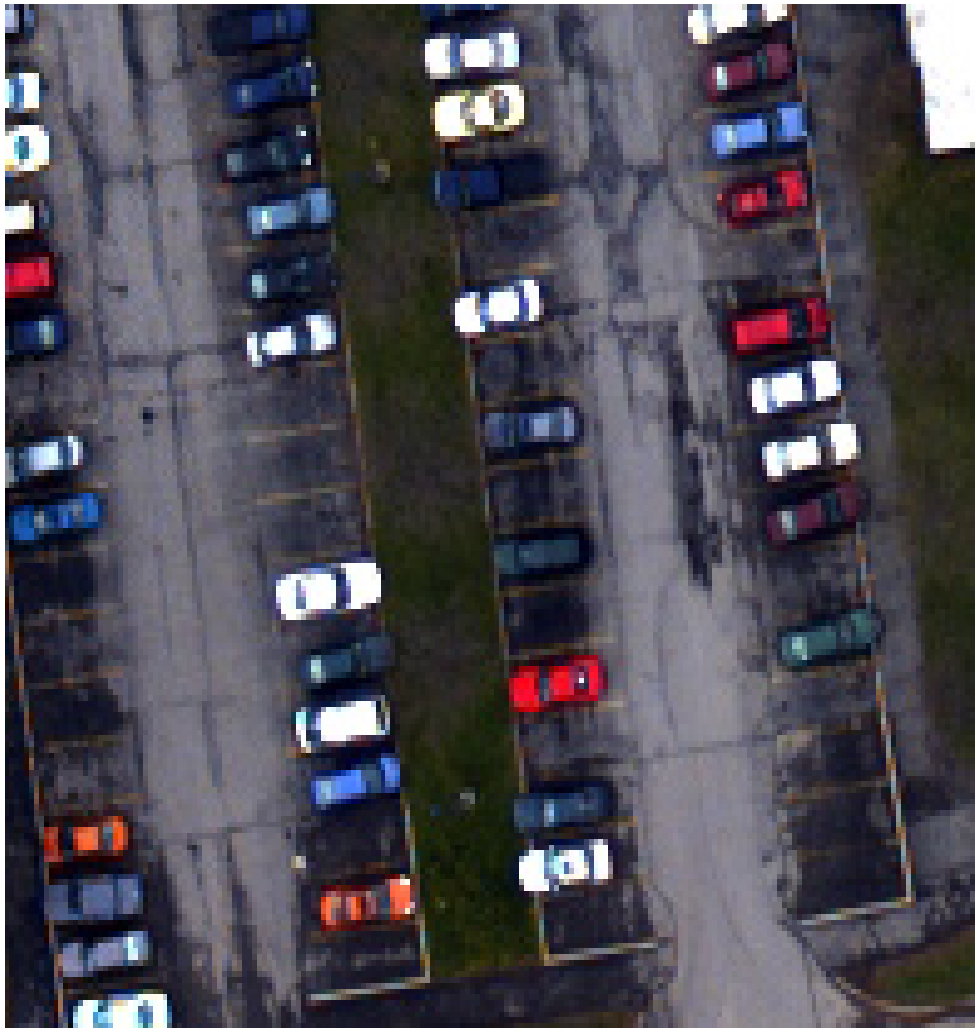
Spatial resolution is often described in terms of the dimension of a pixel on the target

- For example 6” resolution usually means a pixel will cover a 6” x 6” square

Spatial Resolution Comparison – 10”

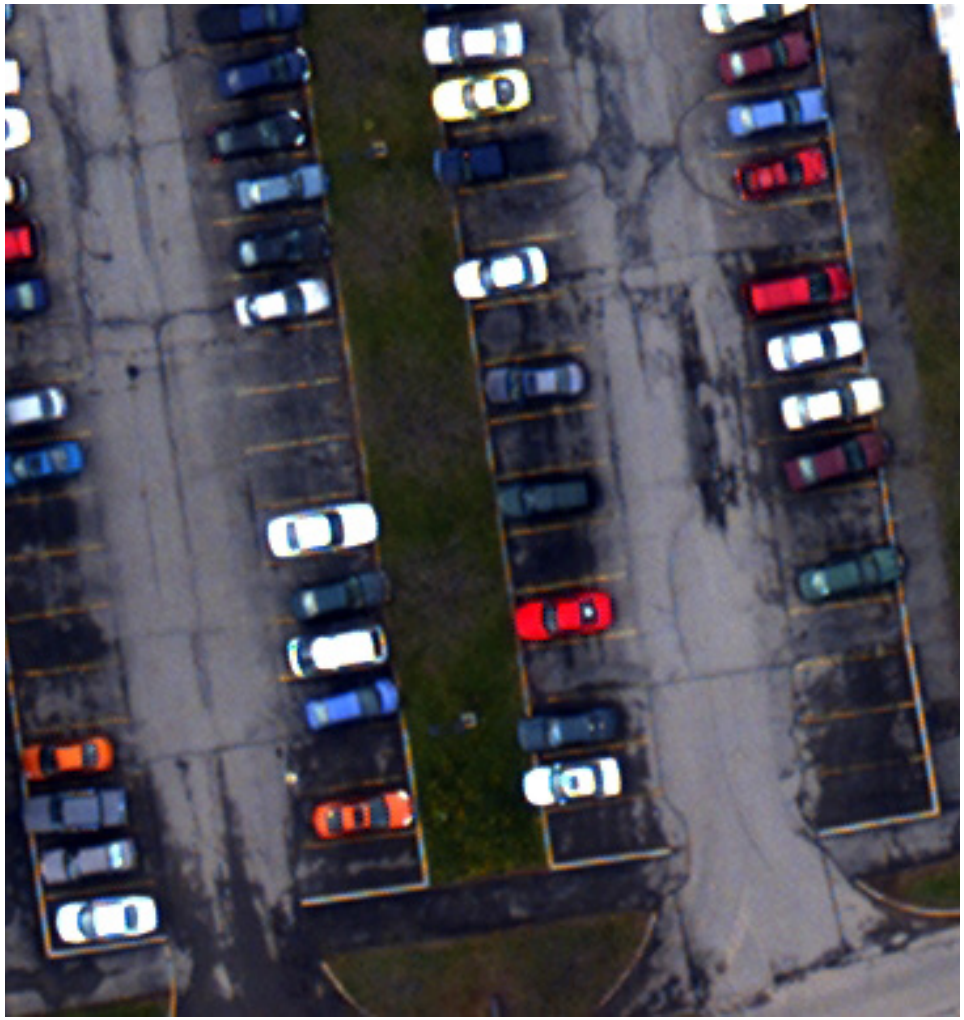


5” Resolution

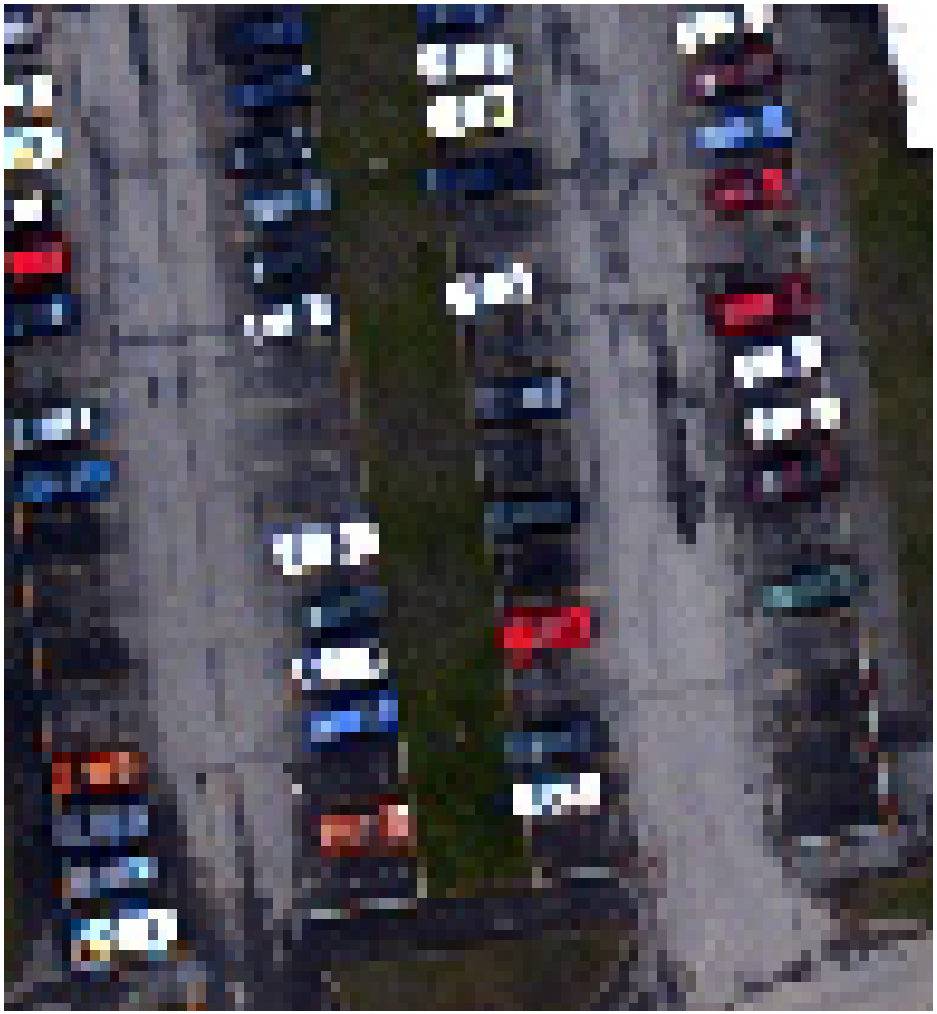


10” Resolution

Spatial Resolution Comparison – 20”

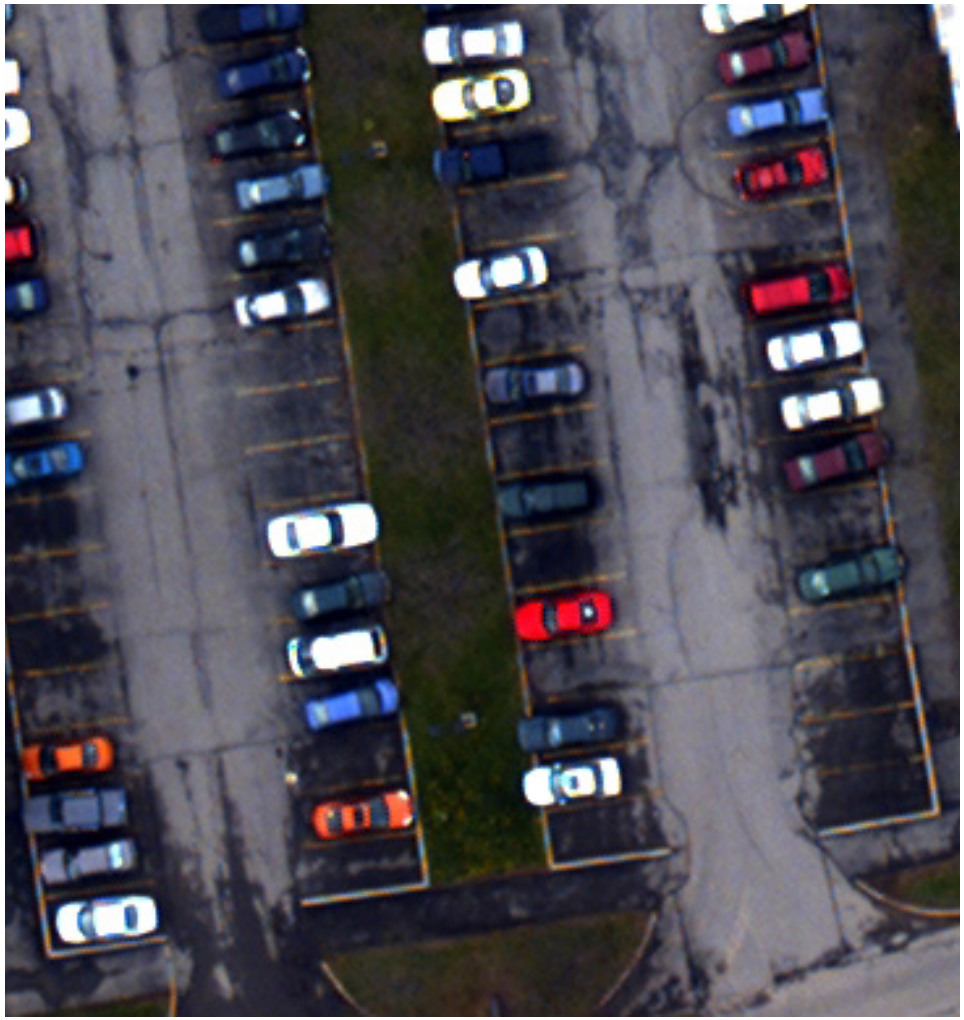


5” Resolution



20” Resolution

Spatial Resolution Comparison – 40”

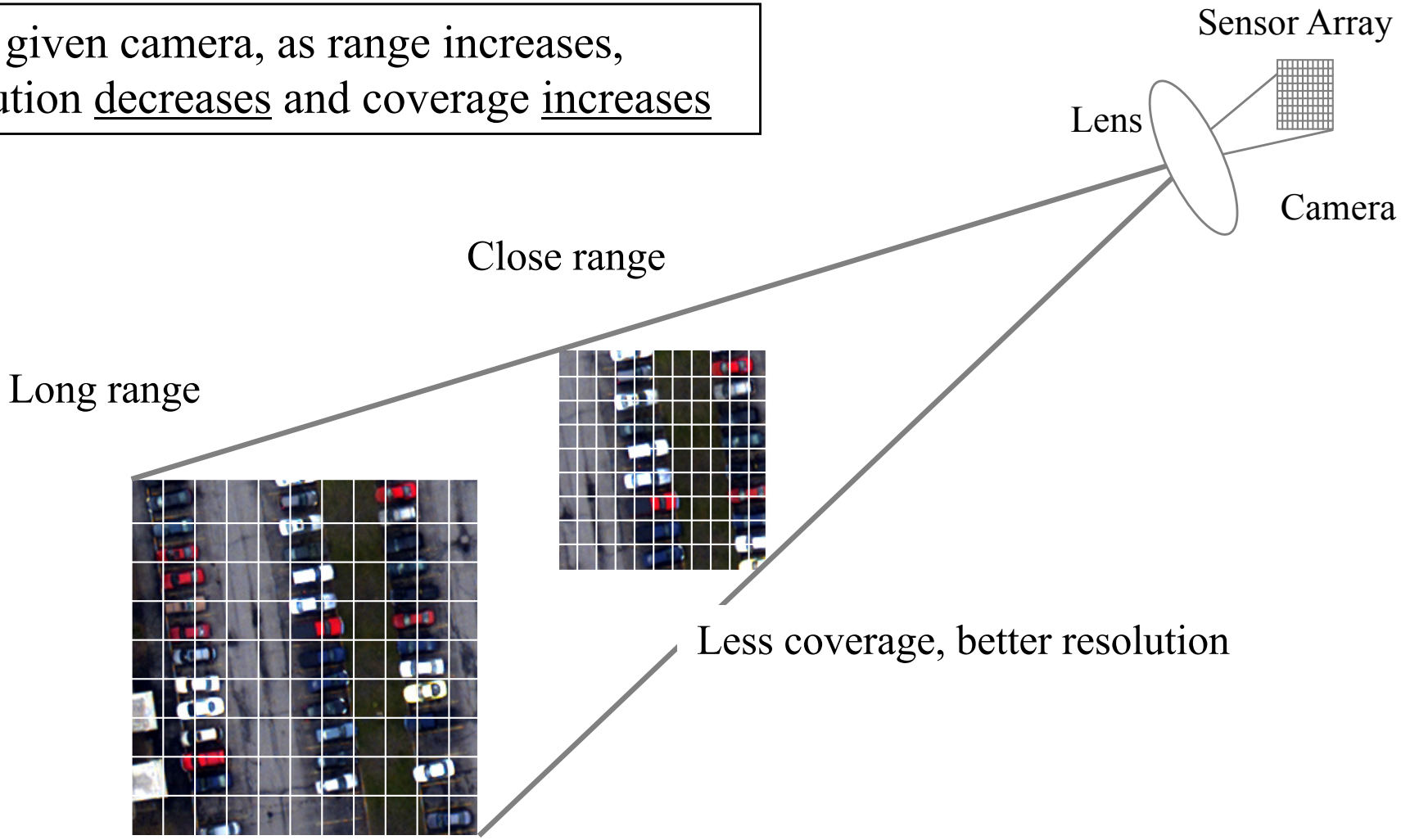


5” Resolution

40” Resolution

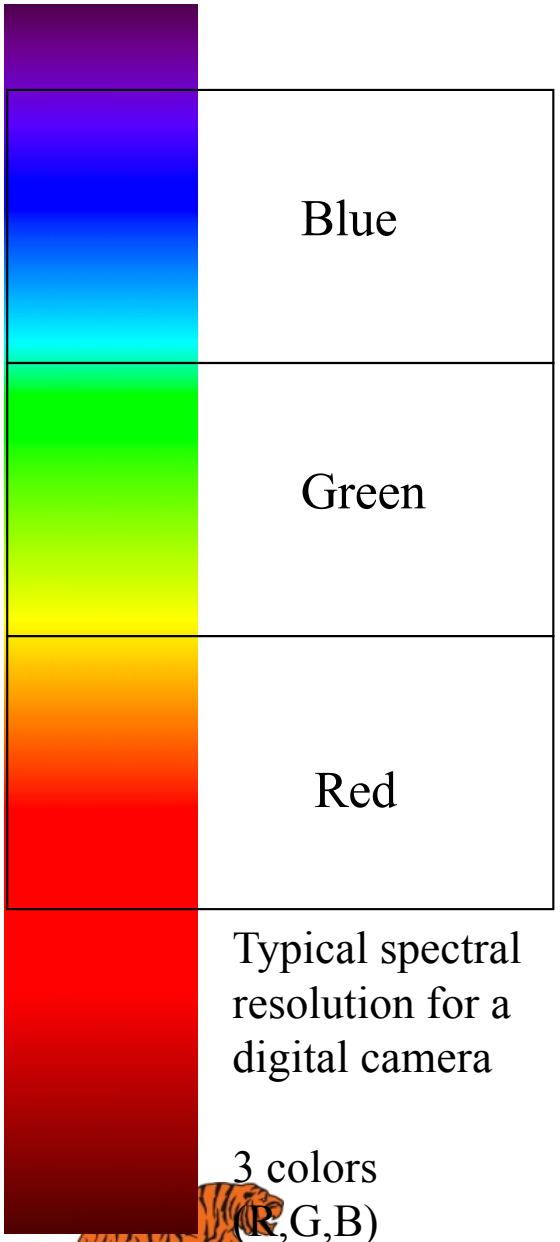
Range Drives Resolution and Coverage

For a given camera, as range increases, resolution <u>decreases</u> and coverage <u>increases</u>



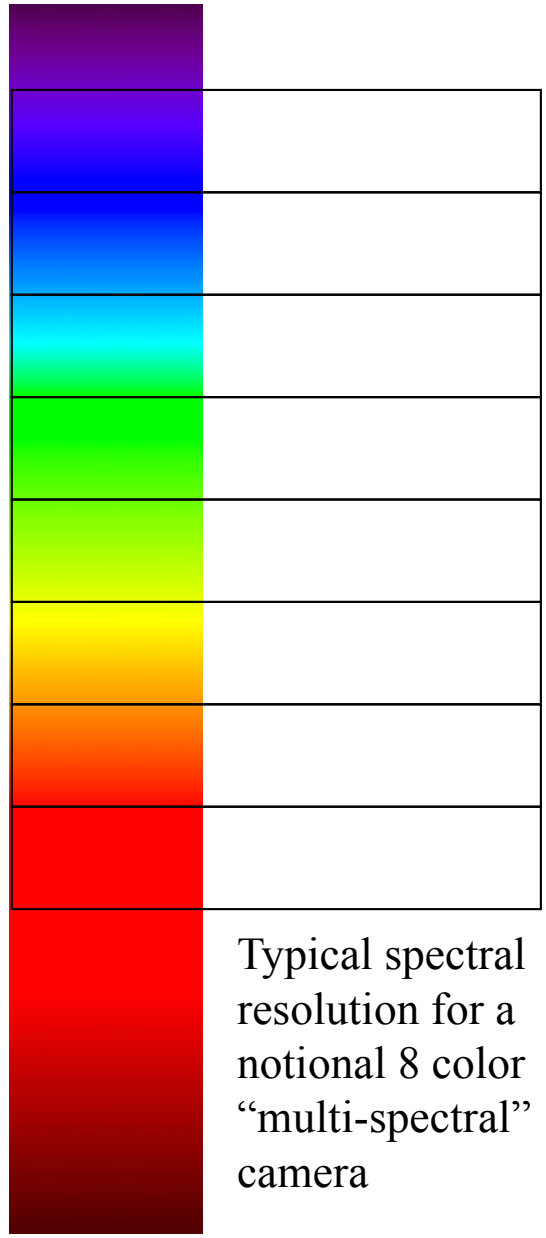
Spectral Resolution

Spectral Resolution

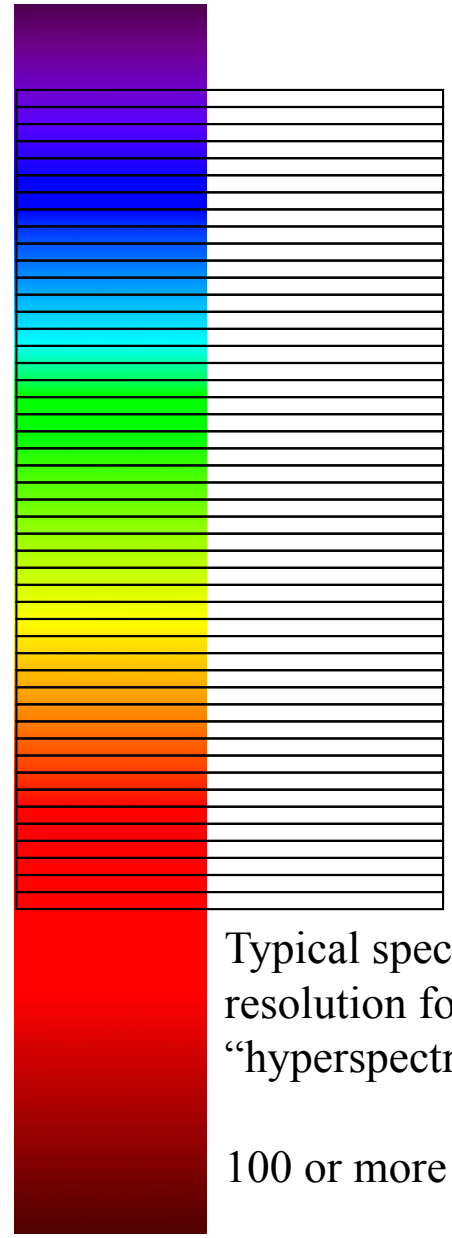


Typical spectral resolution for a digital camera

3 colors (R,G,B)



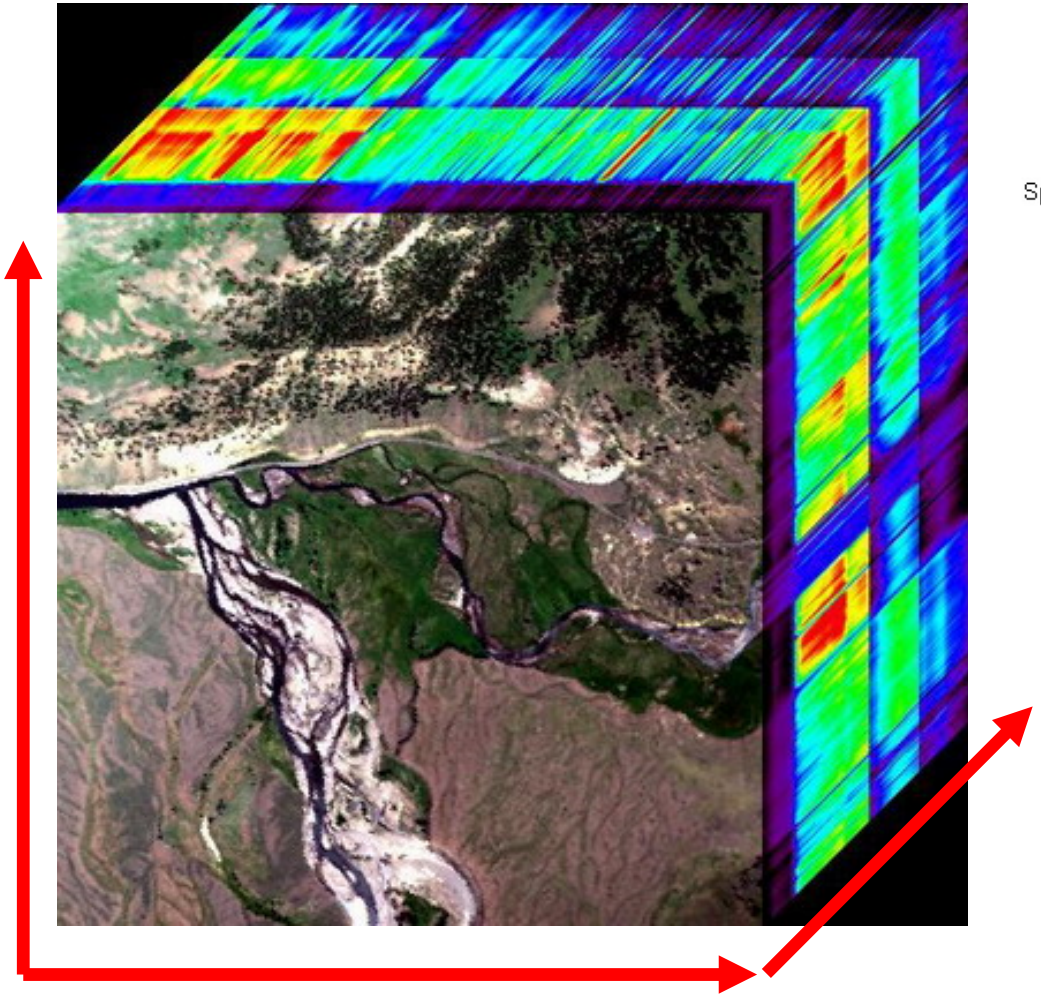
Typical spectral resolution for a notional 8 color "multi-spectral" camera

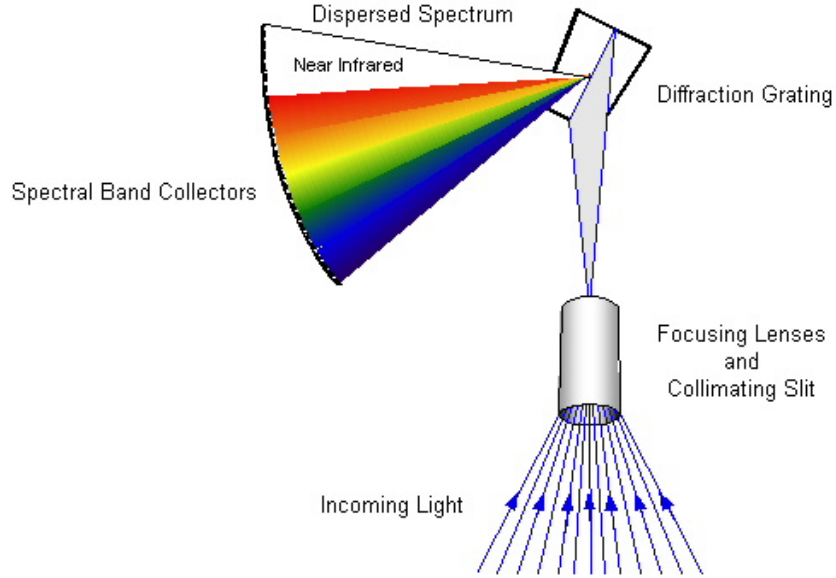


Typical spectral resolution for a "hyperspectral" camera

100 or more colors

Hyper / MultiSpectral Imaging





3 dimensional "image cube"

- every 2 dimensional pixel is represented by a range of color values in the third dimension

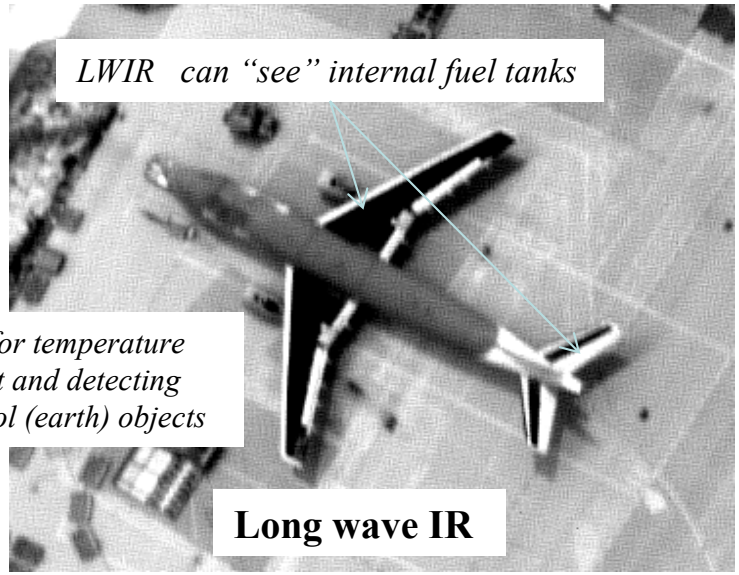
Benefits of Higher Spectral Resolution

- High spectral resolution enables the detection of spectral features of interest and material identification

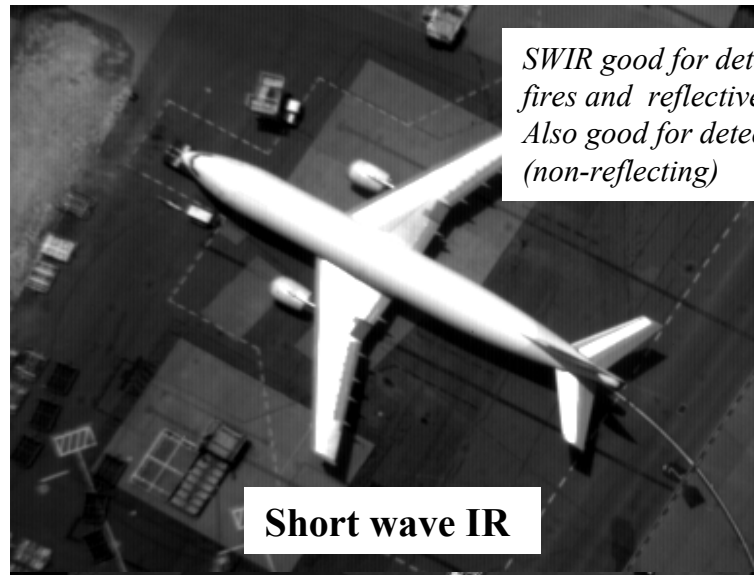
Spectral Phenomenology

- Different spectral regimes can provide various types of information
- Visible/Near Infrared (VNIR)
- Shortwave Infrared (SWIR)
- Midwave Infrared (MWIR)
- Longwave infrared (LWIR)

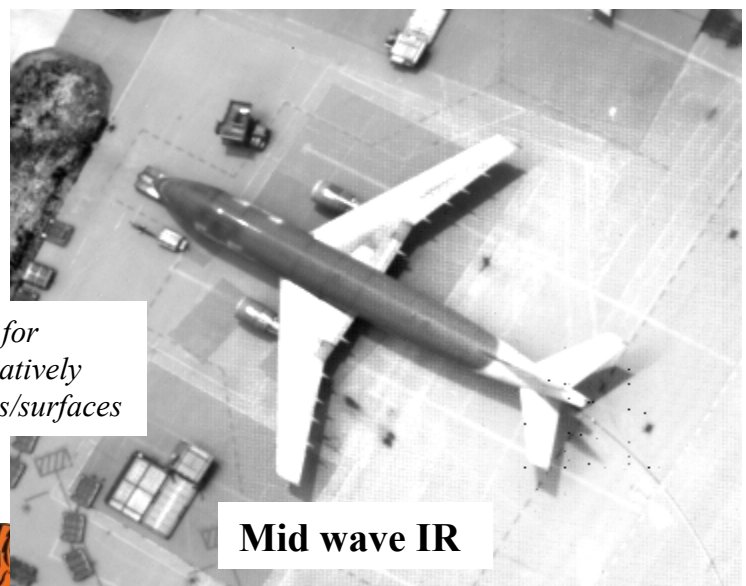
Multi-Band Imagery Example



LWIR good for temperature measurement and detecting relatively cool (earth) objects



SWIR good for detecting hot fires and reflective materials. Also good for detecting water (non-reflecting)



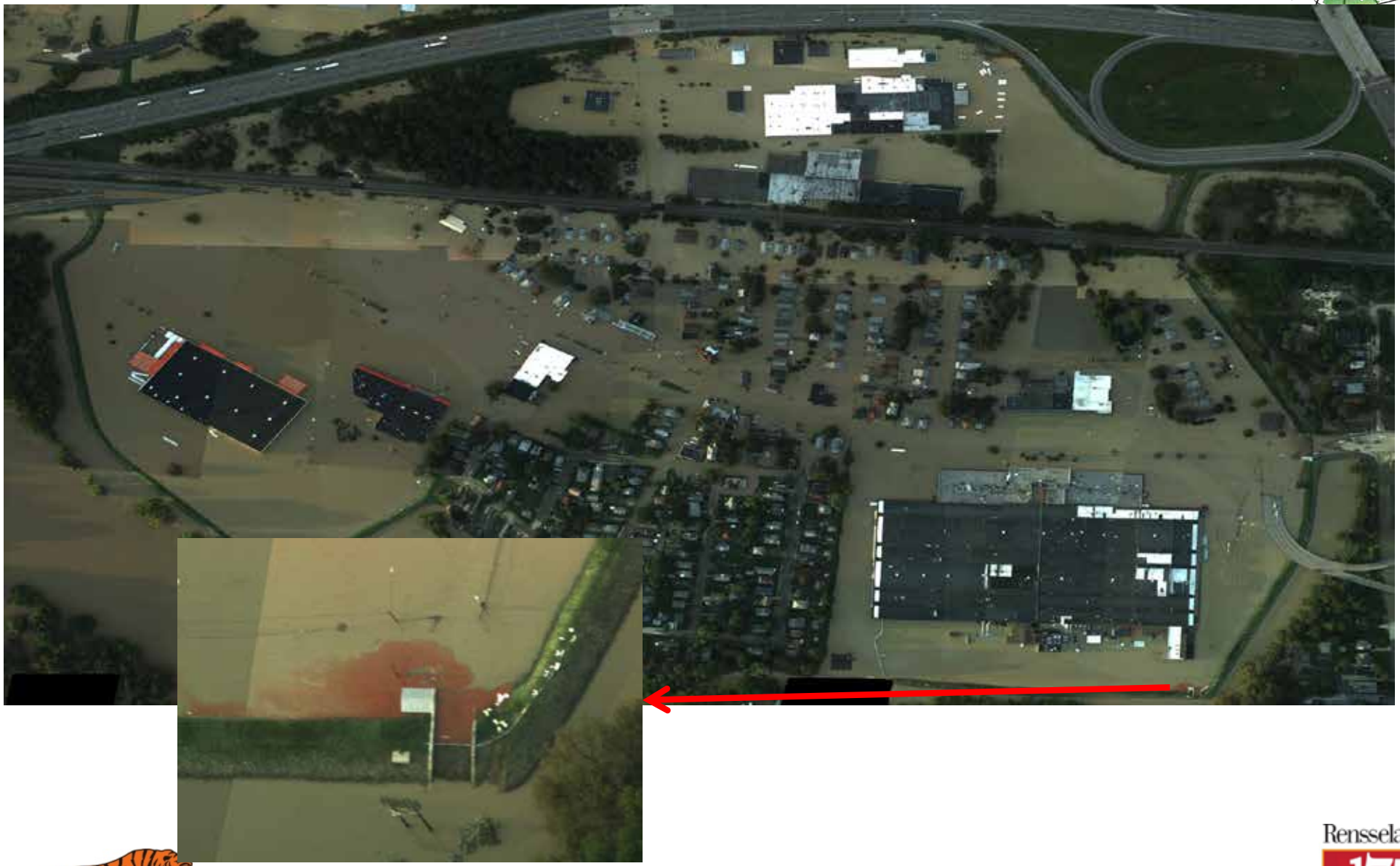
MWIR good for detecting relatively warm objects/surfaces

Vis is good for very high resolution

VNIR Imagery Applications

- VNIR is the most common modality
- Key applications
  - Damage assessment
  - Mapping
  - Floods
- Advantages
  - Readily available and relatively low cost
  - Interpretability is very intuitive - basically “sees” what humans see
  - Very high spatial resolution
- Limitations
  - Good only in daylight
  - Can be obscured by smoke and haze

Flood



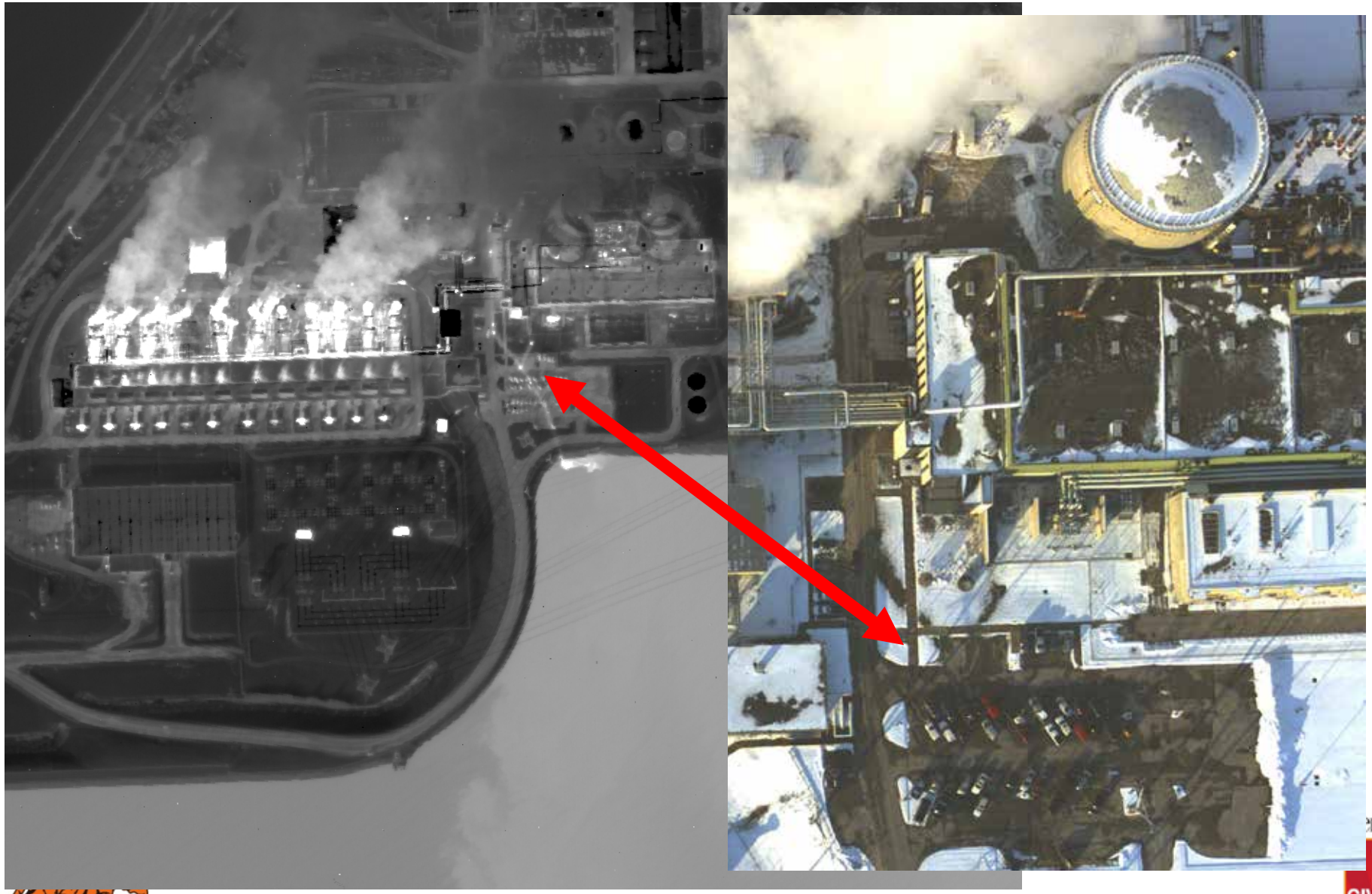
SWIR Imagery Applications

- SWIR
- Key applications
  - Flood mapping
  - Fire detection
- Advantages
  - Good smoke and haze penetration
- Limitations
  - Lower spatial resolution
  - Daylight only
  - SWIR systems are more expensive that VNIR
  - Not as common as VNIR

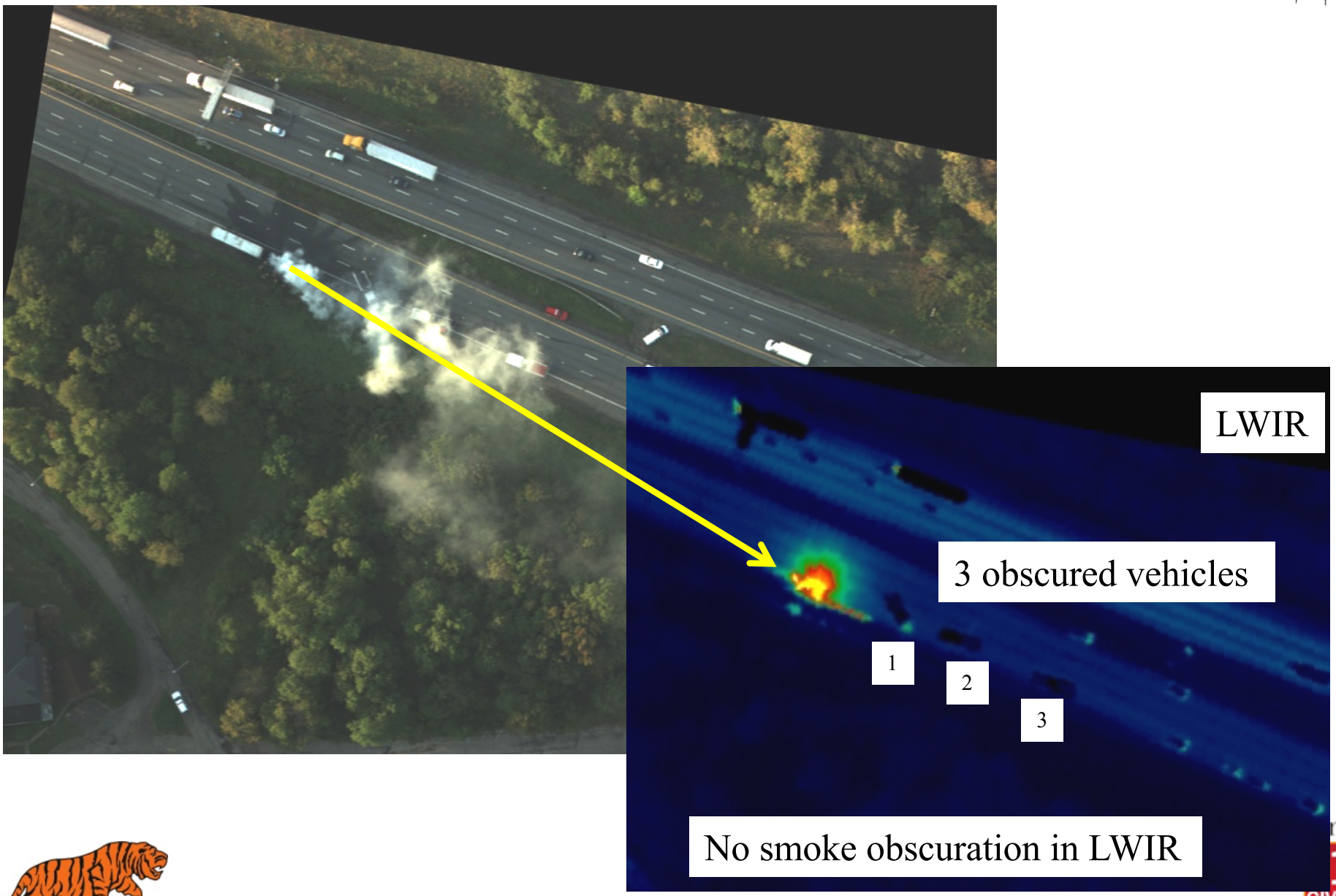
MWIR / LWIR Imagery Applications

- MWIR / LWIR
- Key applications
  - Fire mapping
  - Flood mapping
  - Night time situational awareness
- Advantages
  - Excellent penetration through smoke and haze
  - Day or night
- Limitations
  - Lower spatial resolution
  - MWIR and LWIR systems are much more expensive that VNIR
  - Interpretability requires some knowledge about thermal imaging

Thermal Infrared Imagery



Fire



Temporal Resolution

Video Imaging

- Video imagery is extremely valuable for dynamic scenes
  - looking at relatively small areas over time
- Best suited for rotary wing platforms
  - Allows camera to “dwell” over an area
- Not usually used for mapping
  - It can be done but not typical

“Persistent” Imaging Systems

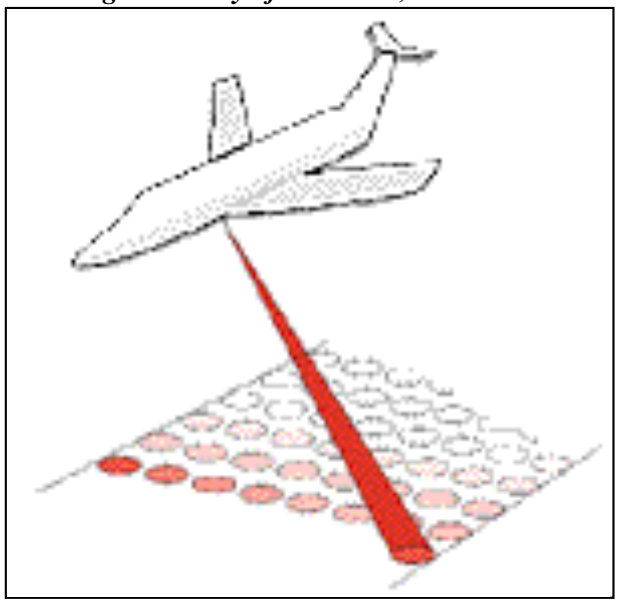
- New high end persistent surveillance systems combine broad area coverage capability of still frame with motion capabilities of video
  - multiple high resolution frame cameras firing at 2 to 5 frames /sec combined together to cover a large field of view
  - imagery is geo-referenced to allow direct GIS insertion

- Example: CORVUS EYE (Harris), Hawkeye II (PSS), etc.

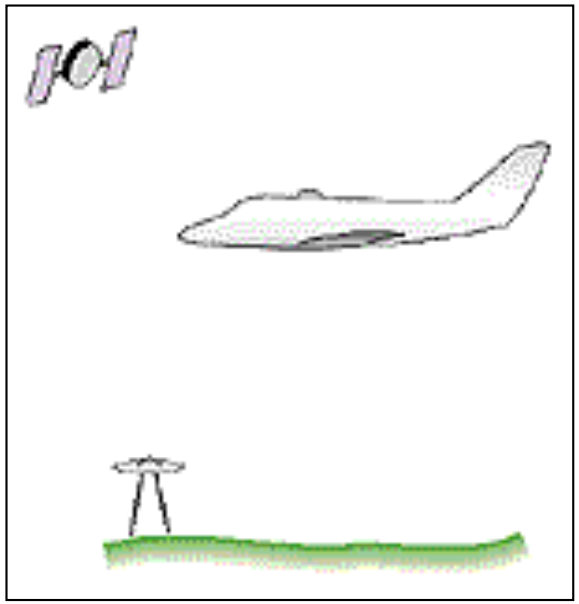
“NON-TRADITIONAL” IMAGING

LiDAR Sensor for 3D Measurements

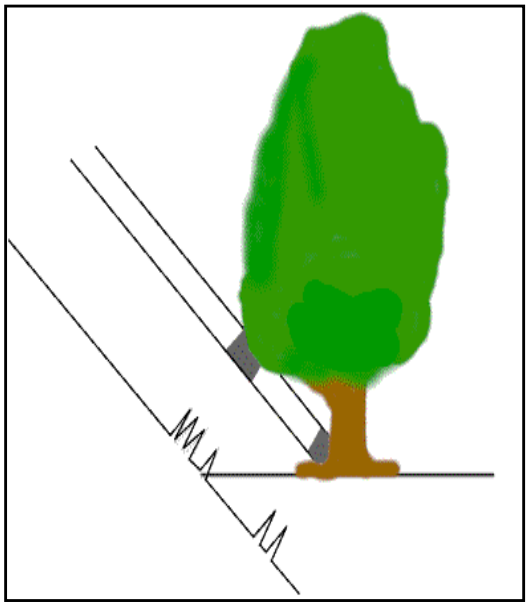
Images courtesy of OPTECH, Inc.



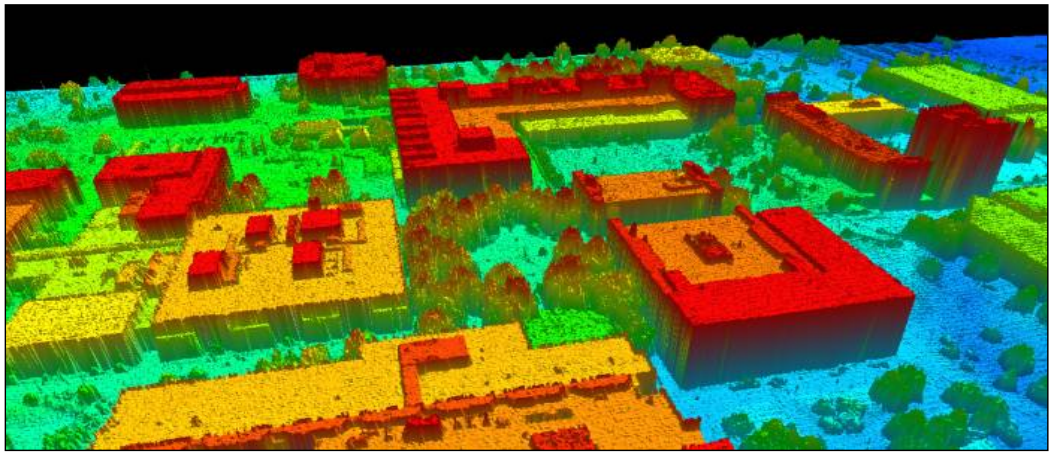
Laser pulses



Geo-located



Multiple returns

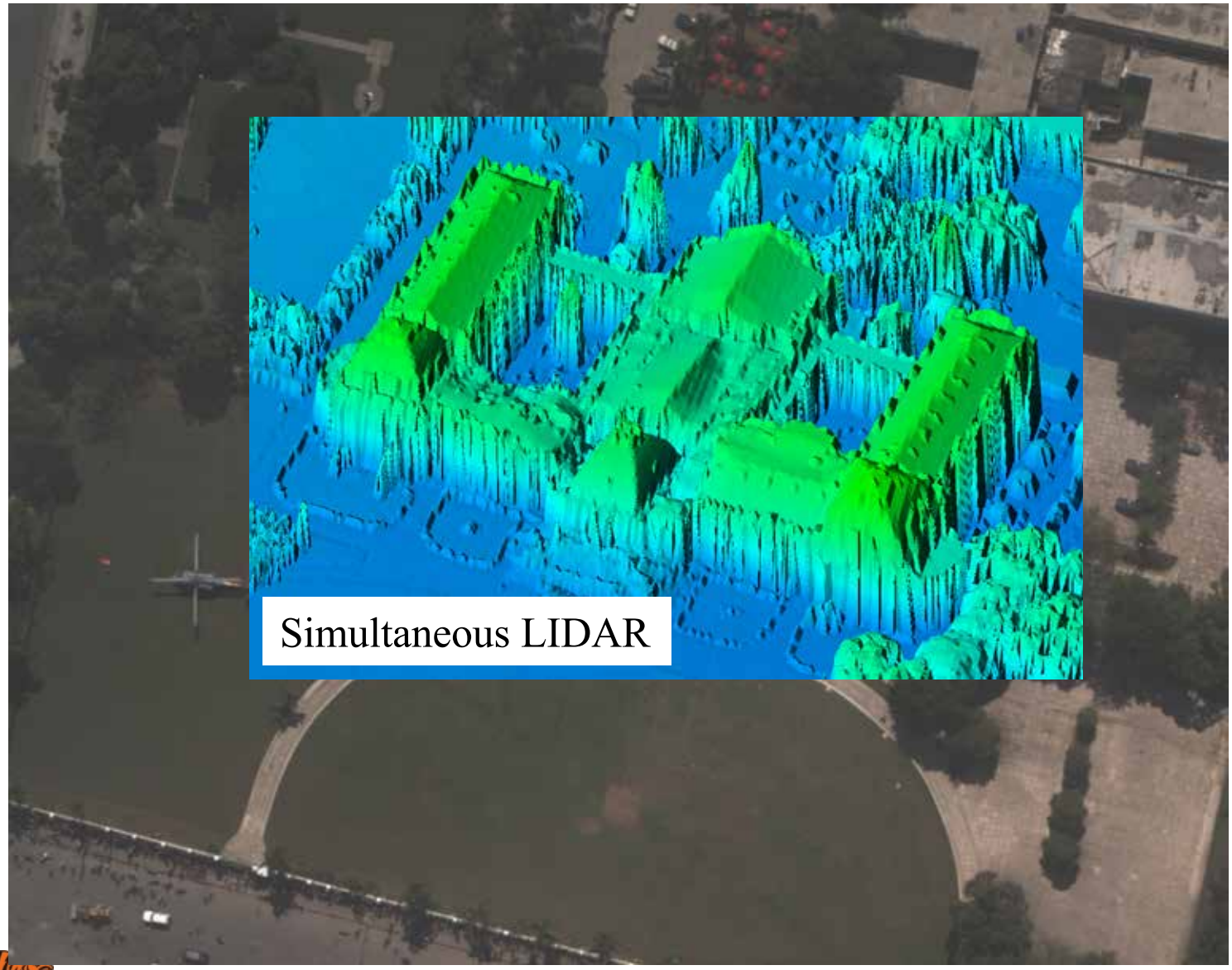


RIT Campus

LiDAR Applications

- LiDAR
- Key applications
  - Terrain elevation mapping
  - Structural measurement
- Advantages
  - Day or night operation
  - Highly accurate 3D measurement
- Limitations
  - Point cloud requires processing and interpretation
  - LiDAR systems are very expensive to procure / operate

Earthquake



Lidar 3D Surface from Point Clouds

- Lidar generates a 3D “point cloud” that is then processed into a 3D surface map

Synthetic Aperture Radar

- SAR systems can operate day or night and in all weather (can "see" through clouds)
- Emerging technology for commercial applications
- Good for flood mapping

SAR Applications

Remote Sensing Data Sources

Platforms: Air vs Space

- Aircraft and satellites have different capabilities and limitations
- Aircraft
  - Pros:
    - flexible deployment (fly under clouds, on-demand, change mission parameters)
    - very high resolution
    - flexible sensor capabilities (e.g. thermal, hyperspectral, lidar)
  - Cons:
    - relatively long time to cover large areas
    - subject to local restrictions
- Satellites
  - Pros:
    - predictable, reliable deployment
    - not subject to local restrictions
    - rapid coverage of large areas
  - Cons:
    - access is limited by orbit and weather
    - sensor technologies limited to what's available

Airborne Data Collection

- Fixed Wing
  - Basic description
  - Advantages
  - Disadvantages
- Helicopter
  - Basic description
  - Advantages
  - Disadvantages
- UAVs
  - Basic description
  - Advantages
  - Disadvantages

Fixed Wing

- Single engine or twin engine aircraft
- Rapid deployment capability
- Lower cost than helicopter
- Aerial mapping based on flightline-based (grid) geometry
- 1,000 minimum altitude restriction
- Minimum weather and visibility restrictions
- Minimum airspeed required

Helicopter

- Highly maneuverable
  - No minimum altitude limitation
- Can hover / dwell over a fixed location
  - No minimum airspeed requirement
- Expensive to operate
- Slower speed for deployment
- Less restrictive weather / visibility than fixed wing

Unmanned Aerial Systems (UAS)

- UASs are rapidly gaining in capability and accessibility
- Relatively low speed
- Extremely flexible
- Very low cost
- Payload weight limits the available sensors
  - Offset by ability to get closer
- Can be used in conditions too hazardous for manned platforms

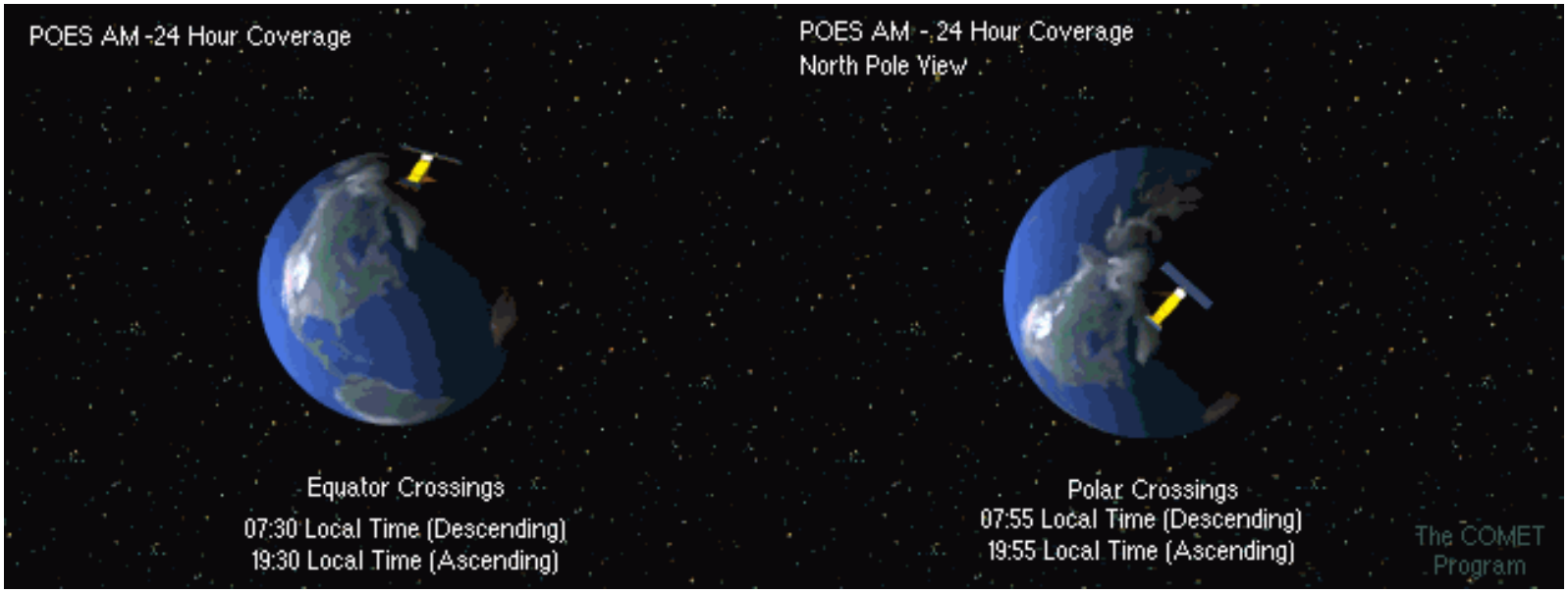
Sources of Satellite Data

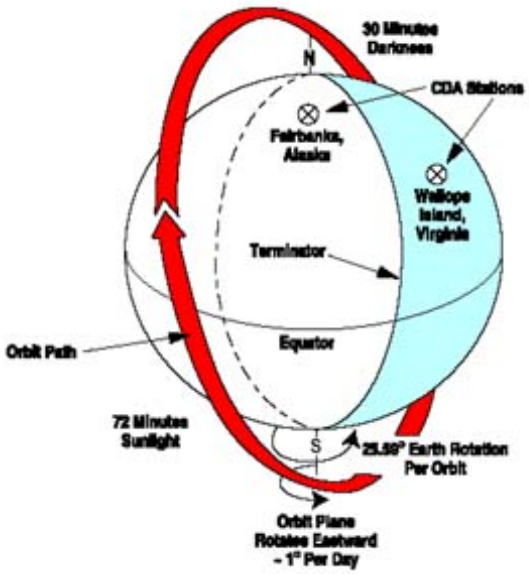
- A number of satellite imagery products are available
- Basic trades between resolution, coverage, and revisit times
- Best solution may involve using multiple satellite systems

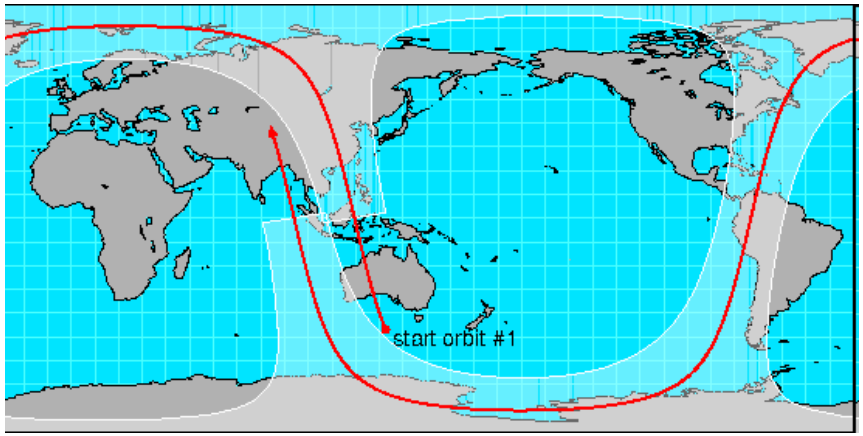
Satellite	Spatial Res (pan)	Spatial Res (color)	Spectral Res (bands)	Coverage	Time	Revisit (nadir)	Revisit (oblique)
GeoEye	0.41m	1.65m	4	15.2km	1030	8 days	3 days
Ikonos	0.82m	3.2m	4	11.3km	1030	5 days	3 days
WorldView1	0.5m	na	na	17.7km	1030	6 days	2 days
WorldView2	0.5m	1.8m	8	16.4km	1030	4 days	2 days
QuickBird	0.65m	2.6m	4	18km	1000	6 days	3 days
RapidEye	na	5.0m	5	77km	1100	6 days	1 day
Landsat 5	na	30.0m	7	185 km	0945	16 days	16 days
Landsat 7	15.0m	30.0m	7	185 km	1000	16 days*	16 days

* Get a pass from either LS 5 or LS 7 every 8 days

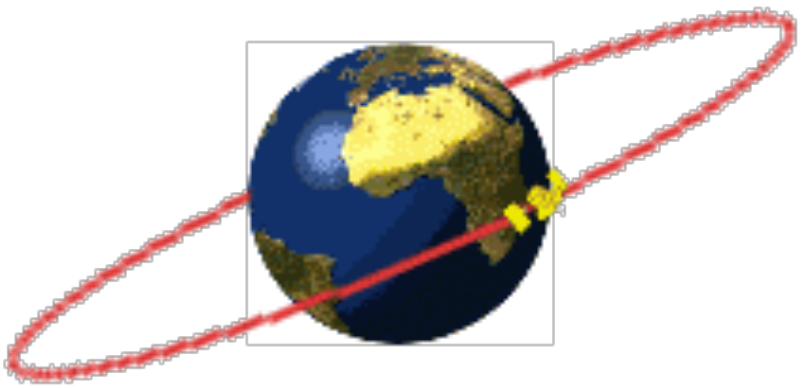
Basic Satellite Operations – Polar Orbit

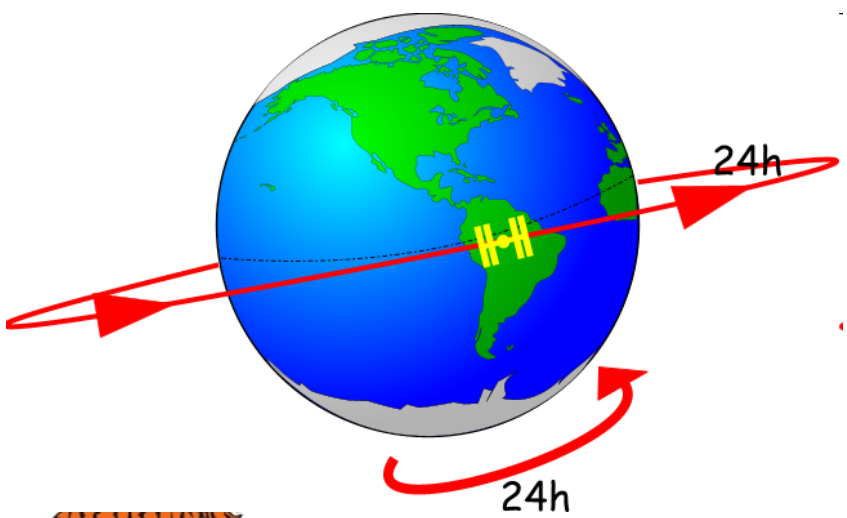


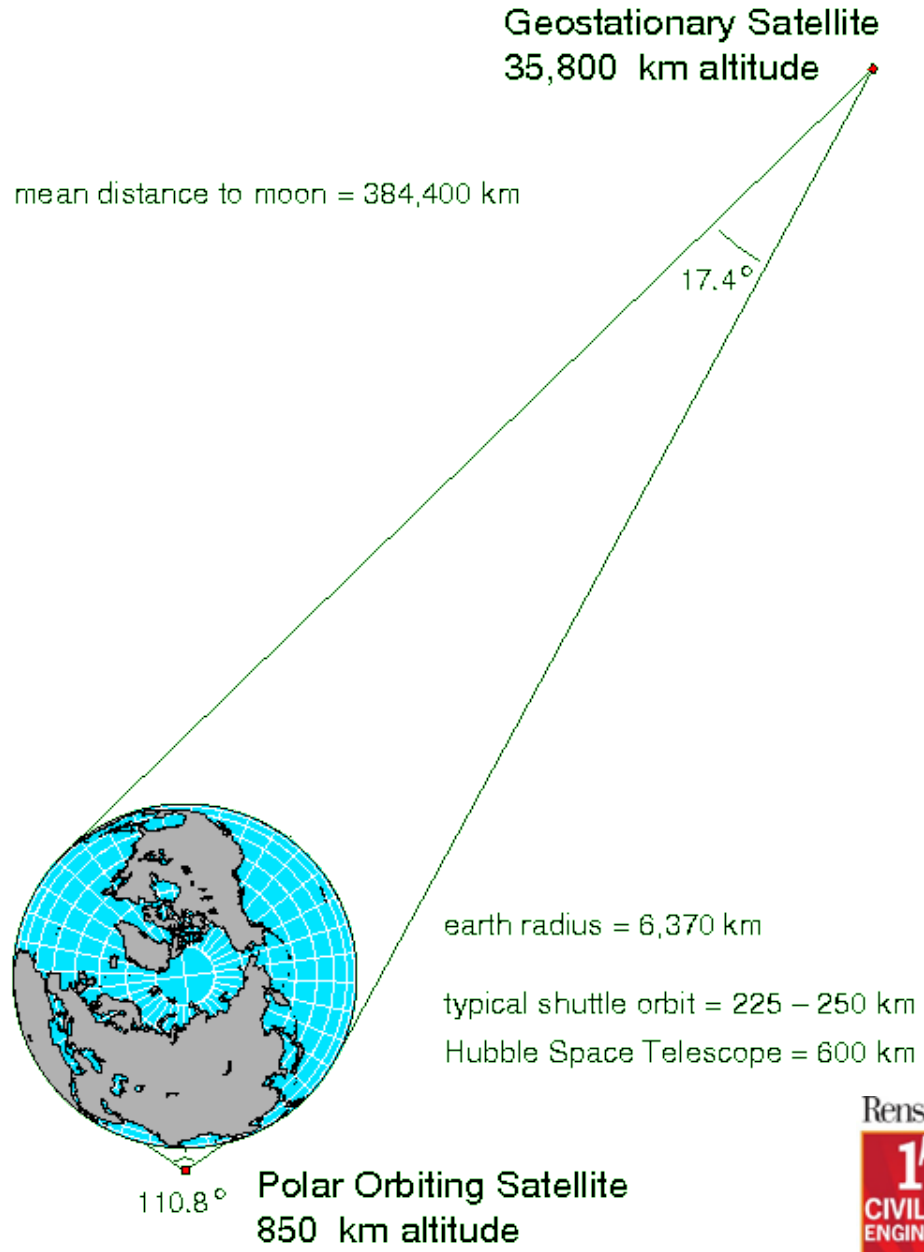




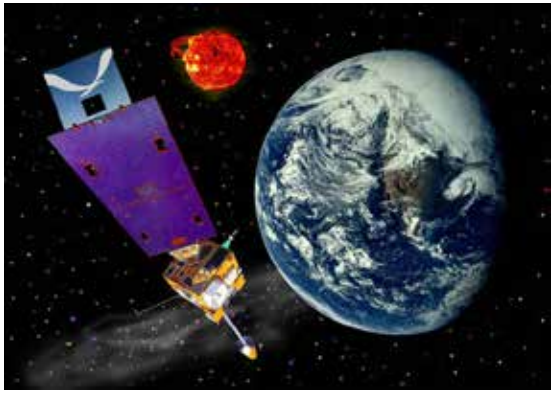
Basic Satellite Operations – Geosynchronous Orbit

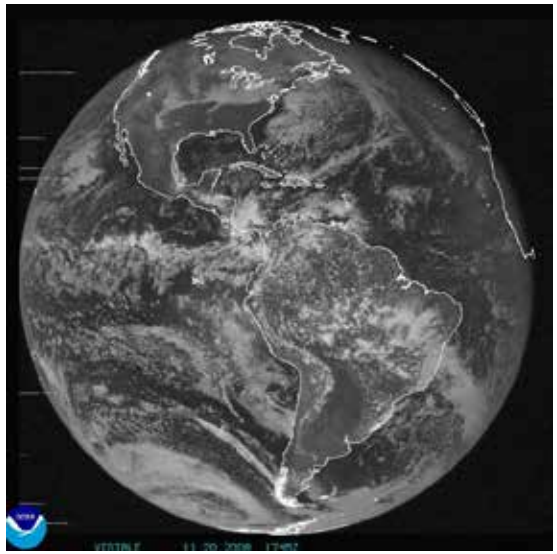


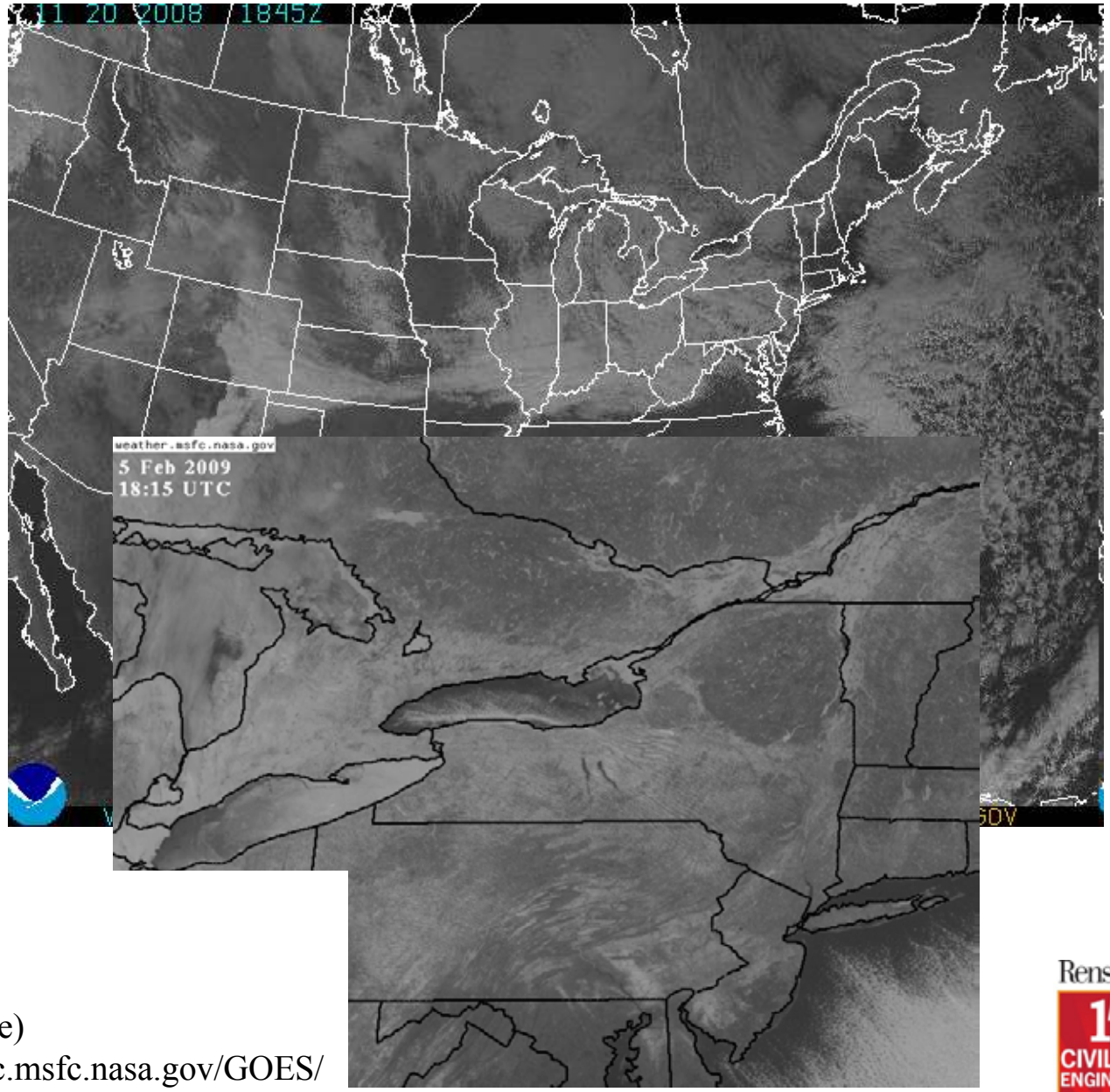




NOAA Weather Satellites (GOES)

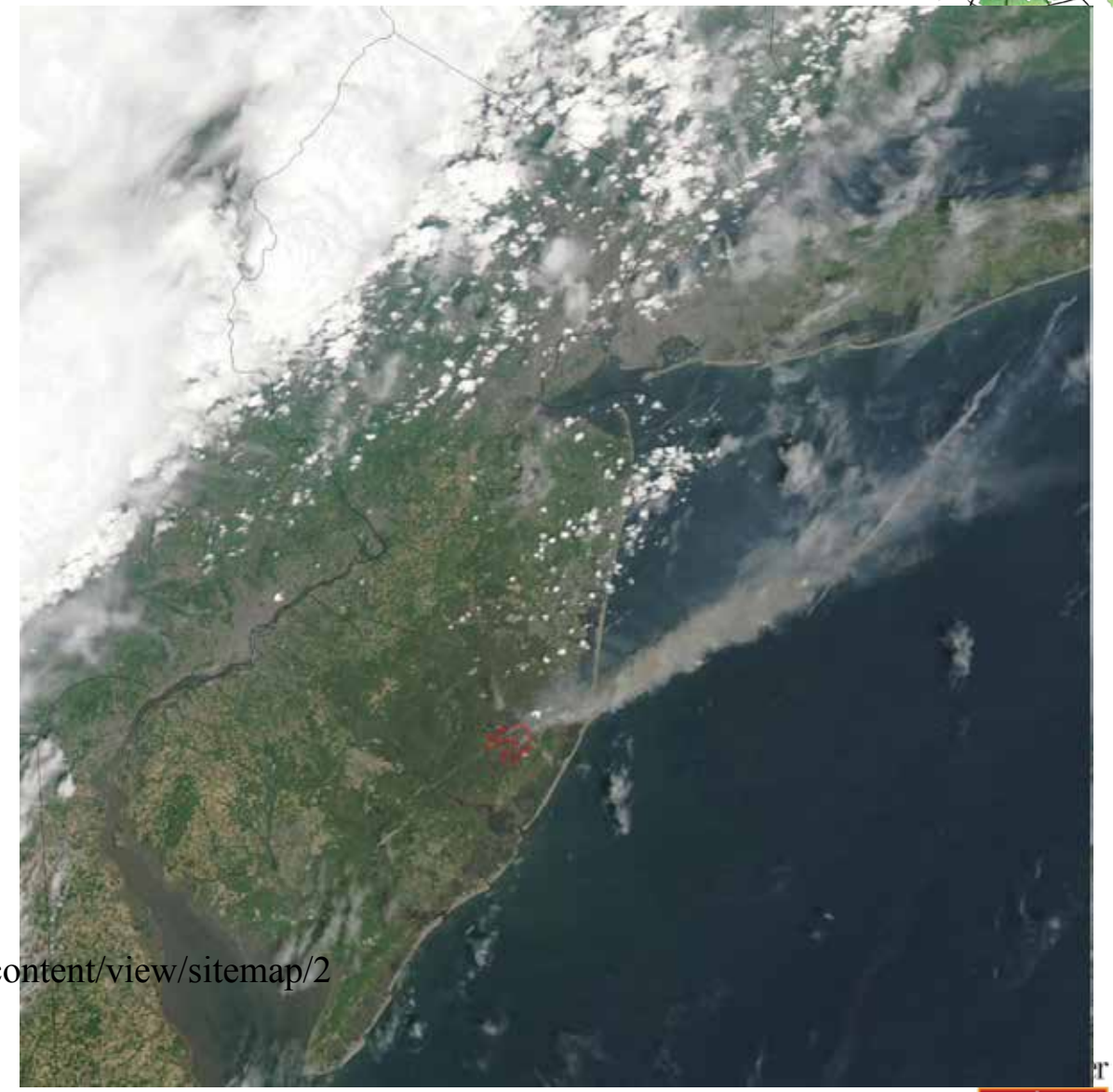






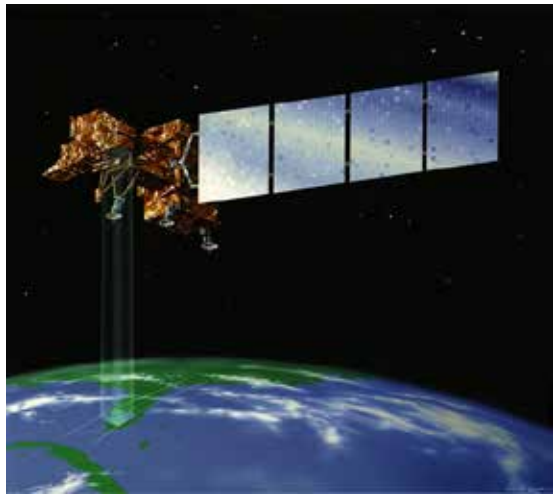
Orbit: Geosynchronous
Resolution: 1 km (vis)
Swath: Full hemisphere
Revisit: Continuous (15 minute update)
Data Available From: http://wwwghcc.msfc.nasa.gov/GOES/

NASA Satellites – MODIS



Orbit: Polar
Resolution: 250 m, 500 m
Swath: 2300 km
Crossing Time: 10:30/1:30
- 2 sats
Revisit Time: 2x day
- 2 sats
Data Available From: https://lpdaac.usgs.gov/content/view/sitemap/2

NASA Satellites – LANDSAT



Orbit: Polar
Resolution: 15 m to 30 m
Swath: 185 km
Crossing Time: 10:00
Revisit Time: 8 days
- Assumes 2 satellites
Data Available From: http://glovis.usgs.gov/

Commercial Satellites - RapidEye

Orbit: Polar
Resolution: 5m
Swath: 15 km
Crossing Time: 10:30
Revisit Time: Daily (constellation of 5 satellites)
Info/Data Available From: http://www.rapideye.net/
- Data is distributed through commercial "partners"

Commercial Satellites – WorldView 2

Orbit: Polar
Resolution: 0.5 m
Swath: 17.6 km
Crossing Time: 10:30
Revisit Time: 5 days
Info/Data Available From: http://www.digitalglobe.com/index.php/5/Welcome+to+DigitalGlobe
- imagery available through commercial "partners"

Other Image Data Sources

- USGS
  - Digital Orthophoto Quadrangles (DOQ) or Digital Ortho Quarter Quads (DOQQ)
    - Digitize aerial photographs corrected for terrain
    - Mapped to UTM coordinates
    - Gray scale or color-infrared with 1 meter resolution
    - http://earthexplorer.usgs.gov/

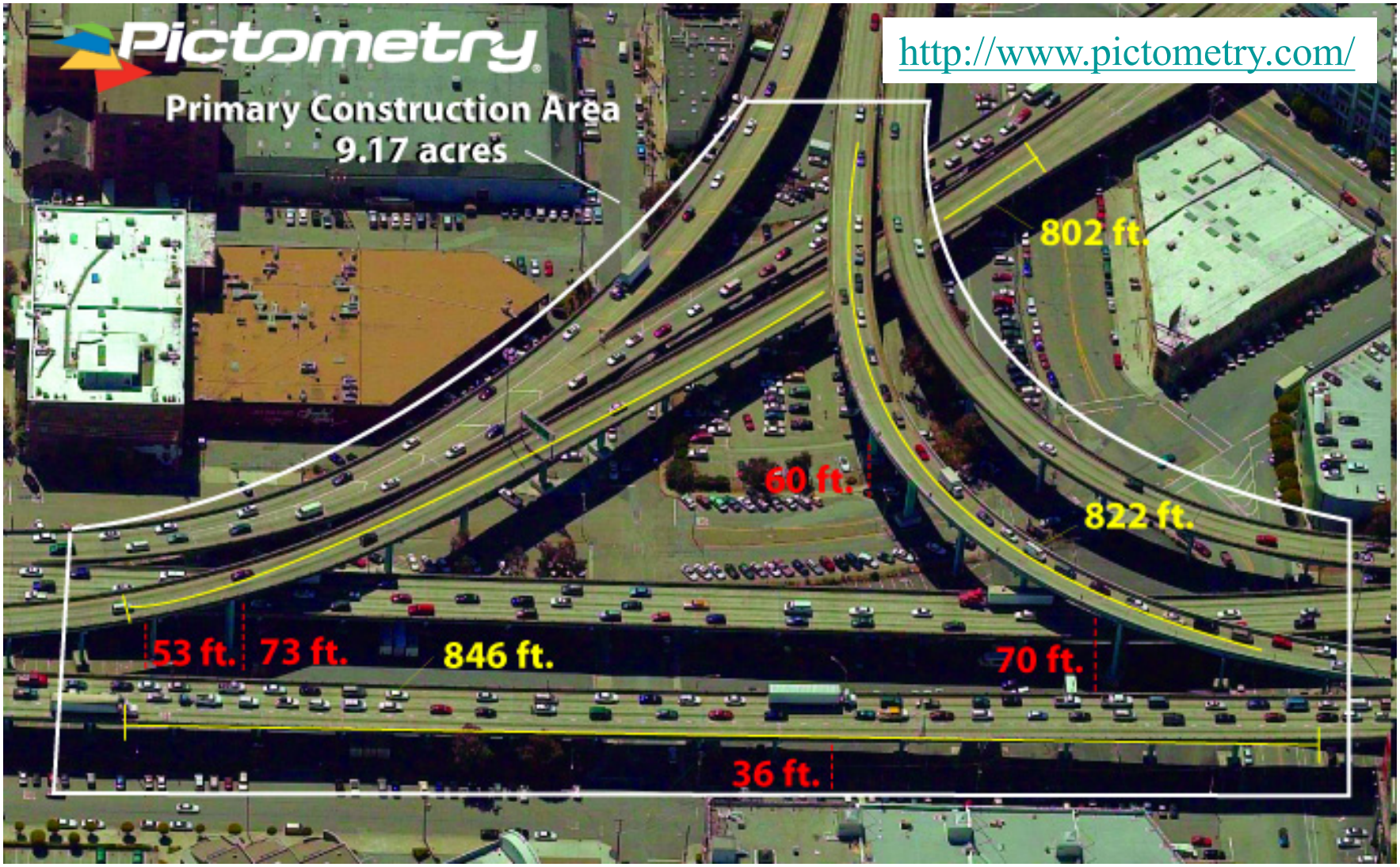
- USDA
  - National Agriculture Imagery Program (NAIP)
    - Digitized aerial photos mapped to UTM coordinates
    - 1 m or 2 m resolution with 5 m or 10 m horizontal map accuracy
    - http://datagateway.nrcs.usda.gov/

- NY STATE GIS Clearing House
  - http://www.nysgis.state.ny.us/gisdata/
  - Access to DOQQ and NAIP imagery

- USGS Seamless National Map Server
  - Excellent site to acquire almost any type of imagery and GIS information
  - Great user interface
  - http://seamless.usgs.gov/

- USGS Hazard Data Distribution System (HDDS)
  - Activated for disasters
  - Portal for a variety of site specific available image data
  - http://hdds.usgs.gov/hdds2/

Aerial Imagery: Pictometry



Imagery can be viewed at bing.com/maps ("Bird's Eye" view)

IMAGERY VIEWING TOOLS

Other Tools

- Pictometry – Electronic Field Study
  - Excellent measurement tool and oblique image viewer
  - Common tool in many county government GIS departments
- ArcView/ArcMap
  - Great for map based analysis and GIS integration
  - Image viewing is relatively clumsy but feasible
  - Fairly standard in GIS organizations
- Photoshop
  - Great for viewing / manipulating imagery
  - Not so good for map based analysis
- ERDAS Imagine
  - Very good for viewing imagery
  - Good GIS integration
  - Requires imaging expertise to be useful
- ENVI
  - Excellent image visualization and manipulation
  - Improving GIS integration
  - Requires imaging expertise to be useful

SCENARIOS

Flooding

					Daylight Only			Day / Night	
Scenario	**Characteristics**	**Essential Elements of Information (EEIs)**	**Spatial Resolution / Point Density**	**Geolocation Accuracy**	**Color**	**MSI/HSI**	**SWIR**	**MWIR / LWIR**	**LiDAR**
Flood	***Innundated roads and facilities. Debris deposits on roads. Washouts, undermined roads. Damage to bridges and bridge approaches. Hazardous effluents from innundated storage or transmission facilities.***	Location of current innundation boundaries	0.5m	2.0m	X	X	X		X
		Location of high water boundaries	0.1m vert, 1.0 m horiz	2.0m	X	X			
		Debris location and passability	0.2m	1.0m	X				X
		Washout of roads location and severity	0.1m	1.0m	X				X
		Undermining of roads location and severity	0.1m		X				X
		Presence of effluents in flood waters	0.5m	2.0m	X	X		X	
		Presence of effluent residue	0.5m		X	X			
		Identification / quantification of effluents	1.0m	2.0m		X			
		Bridge damage assessment	0.1m		X				X

Flooding – Operational Considerations

- Low clouds / poor visibility
- Early deployment (during high water) to detect effluents and map inundation
- Side oblique is useful for high water mark measurement on sides of buildings
- Reference up-to-date pre-flood imagery to see what is under water
- Flood types – storm surge/tsunami, riverine

Earthquake

					Daylight Only			Day / Night	
Scenario	**Characteristics**	**Essential Elements of Information (EEIs)**	**Spatial Resolution / Point Density**	**Geolocation Accuracy**	**Color**	**MSI/HSI**	**SWIR**	**MWIR / LWIR**	**LiDAR**
Earthquake	***Building damage / collapse. Bridge damage / collapse. Roadway cracks / faults. Roadway debris blockage. Structural fires. Effluents from damaged tanks/ transmisison lines.***	Building damage assessment	0.2m	1.0m	X				X
		Debris location and passability	0.2m	1.0m	X				X
		Presence of effluents on surfaces	0.5m	2.0m	X	X		X	
		Fire location and severity	0.5m	2.0m			X	X	
		Bridge damage assessment	0.1m	1.0m	X				X

Earthquake – Operational Considerations

- Loss of infrastructure / communications
- Obscuration (smoke)
- Oblique imagery useful for building damage assessment
-

Tornado/Wind

					Daylight Only			Day / Night	
Scenario	**Characteristics**	**Essential Elements of Information (EEIs)**	**Spatial Resolution / Point Density**	**Geolocation Accuracy**	**Color**	**MSI/HSI**	**SWIR**	**MWIR / LWIR**	**LiDAR**
Torndao / High Wind Event	***Scattered debris, possibly overturned vehicles, downed power lines and poles, focused on relatively small area***	Location of damage boundaries	0.5m	2.0m	X				X
		Debris location and passability of roads	0.2m	1.0m	X				X
		Location of downed lines and utility poles	0.1m		X				X
		Location and severity of fires	0.3m	1.0m	X		X	X	
		Building damage assessment	0.1m		X				X

Tornado / High Wind Operational Considerations

- Event / recovery may be at night
- Presence of other storm cells in area

Man-made (Explosive)

					Daylight Only			Day / Night	
Scenario	**Characteristics**	**Essential Elements of Information (EEIs)**	**Spatial Resolution / Point Density**	**Geolocation Accuracy**	**Color**	**MSI/HSI**	**SWIR**	**MWIR / LWIR**	**LiDAR**
Man-made (Explosive)	***Scattered debris, possibly overturned vehicles, downed power lines and poles, large fires and heavy smoke (possibly toxic) focused on relatively small area. Large congregation of victims / bystanders.***	Location of damage boundaries	0.5m	2.0m	X				X
		Debris location and passability of roads	0.2m	1.0m	X				X
		Location and quantification of victims / bystanders	0.1m		X				X
		Location and severity of fires	0.3m	1.0m	X		X	X	
		Building damage assessment	0.1m		X				X

Man-Made (Explosive) Operational Considerations

- Event / recovery may be at night

APPENDIX F:
Access Restoration Planning User Manual

Rensselaer

CITE

Center for Infrastructure,
Transportation, and the Environment

Access Restoration Planning User Manual

Version 1.0

June 30, 2017

Table of Contents

1. Introduction **2**
1.1 ARP Overview **2**
1.2 ARP System Components **2**
2. ARP Software Installation **3**
2.1 Extracting files to local drive **3**
2.2 Contents of "arp" Directory **4**
2.3 Starting the ARP Software **5**
2.4 Stopping the ARP Software **8**
3. Geospatial Debris Data – Converting Shapefile to GeoJSON format **8**
4. Interface Workflow **11**
4.1 Home Screen - Overview **11**
4.2 Files (Data File Upload & Creation) **11**
4.2.1 Required Columns for Data Files 13
4.2.2 Entry Point and Terminal Point File Creation 14
4.2.3 Using debris data to calculate restoration times 15
4.3 File View - Data File Visualization **16**
4.4 Restorations – Create and Run Restorations **16**
4.4.1 Create Restoration 16
4.4.2 Run Restoration 17
4.4.2 Restoration Results and Visualization 18

1. Introduction

The purpose of this document is to provide an overview, installation instructions, and a step-by-step walkthrough of the Access Restoration Planning (ARP) software.

1.1 ARP Overview

Disaster response is a critical aspect of emergency management. The aim is to provide immediate assistance to maintain life and support affected population in the event of an emergency. Disaster response typically aims to allocating resources and to rehabilitate roads and reach critical points, such as hospitals, of an impacted area. Without proper access, personnel and equipment cannot mobilize impacted areas.

The ARP software is used to provide optimal solutions to restoring access to cities in the event of a natural disaster or any other circumstance that causes mass disruptions. The ARP software incorporates a mathematical model that provides an optimal restoration given a set of parameter values such as life valuation, private costs, resource capacity, optimization routine, and performance measure.

1.2 ARP System Components

The ARP contains three functional components: 1) Data File Upload, 2) Restoration Creation and Optimizer, and 3) Results Visualization. These three components encompass the business flow for generating restoration plans. Below is a high level synopsis of the components of the ARP. Details regarding these components are provided in subsequent sections of this document.

- ***Data File Upload and Creation*** – Specific data is required by the mathematical model in order to provide an optimal restoration solution. The software ingests data files that define road network nodes/links, terminal network nodes, entry points, restoration times, and debris. Data files are associated with a specific city or geographic region. The software also provides users with the ability to view the aforementioned data on an interactive map.

- ***Creating and Optimizing a Restoration –*** A restoration run is created by specifying the data files and parameters to use for a particular scenario. The data and parameters are incorporated into a math model that generates an optimal solution. The software allows users to run and store multiple restoration runs.

- ***Results Visualization and Reporting –*** The results of each restoration are made available upon the completion of each run. The ARP software allows

users to view and interact with the results using a dynamic data dashboard. This dashboard includes a summary of the results (including total restoration time, entry nodes used, terminal points reached, total deprivation cost, and the total Social Cost) as well as a detailed, time-lapse view of results as a function of time. In addition, users can download results can be downloaded for analysis

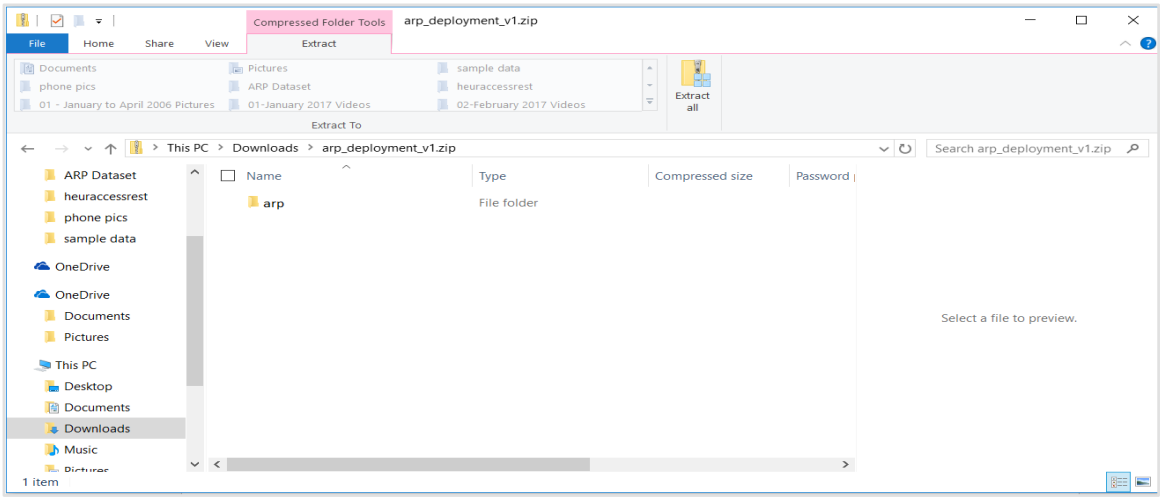
2. ARP Software Installation

The ARP can be installed and used on any Windows-based desktop or laptop. The software is packaged into a zip file named ***arp_deployment_v1.zip***. This zip file contains all required system components including web server, database server, etc. Once downloaded, the user simply needs to extract the files, save them on their local machine, and then launch (or shut down) the software at their convenience.

2.1 Extracting files to local drive

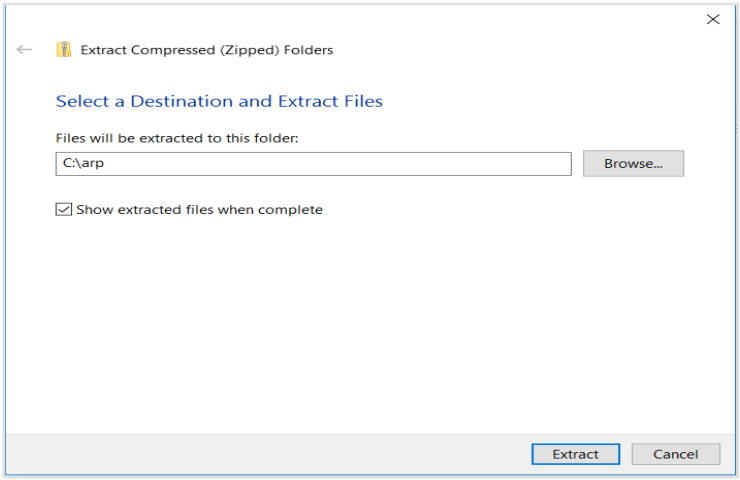
Once you have successfully downloaded the ***arp_deployment_v1.zip***, you will need to extract the files and save them to your local hard drive. This can be accomplished by performing the following steps:

1. Double click on the ***arp_deployment_v1.zip*** folder. You will notice an "arp" file contained within the zip file. You will also notice an "Extract All" icon within the header of the windows explorer dialog box that is open.



2. Select the "Extract All" icon in the dialog (pop-up) box that appears and enter the destination where you would like to store the extracted files. Once you designate a location to store the "arp" package select the Extract button.

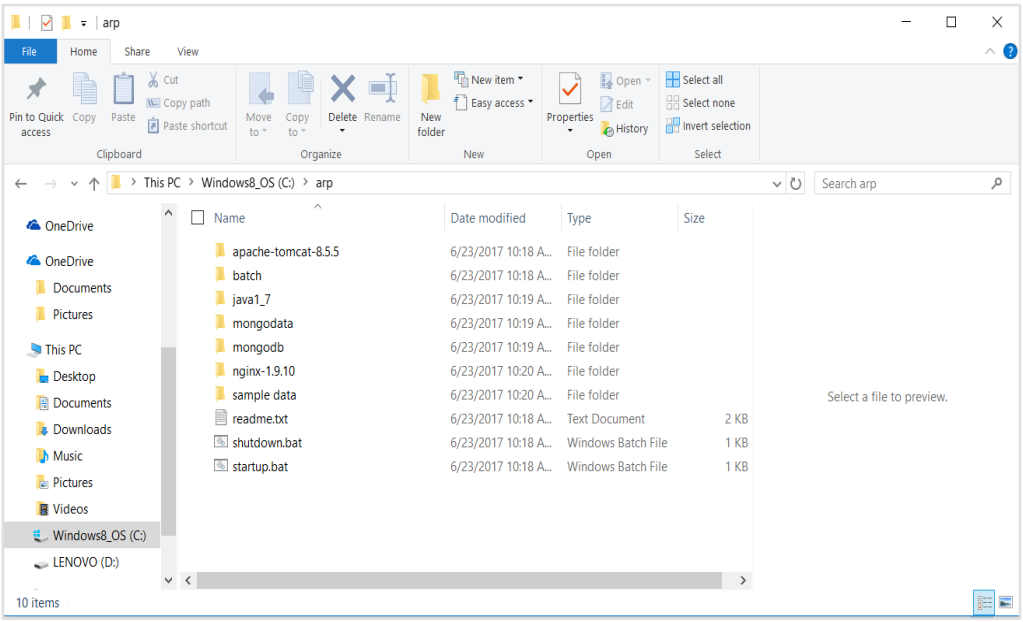
NOTE: You can store the "arp" package anywhere on your computer, however, we recommend that they are stored directly under the C:\ drive as is shown below:



3. Once the files are extracted, you can navigate to the C:\ within Windows Explorer and open the "arp" directory.

2.2 Contents of "arp" Directory

The "arp" directory created on your machine contains information for the web, application, database servers, sample data files, and the ability to start-up and shutdown the ARP software at any time.



1. *Web Server / Application Server / Database Server Folders* – The first six folders focus on the various server and environment information needed to successfully startup the ARP on your machine. You do not need to worry about anything within these folders as follows:
    - apache-tomcat-8.5.5
    - batch
    - java 1_7
    - mongodata
    - mongodb
    - nginx-1.9.10

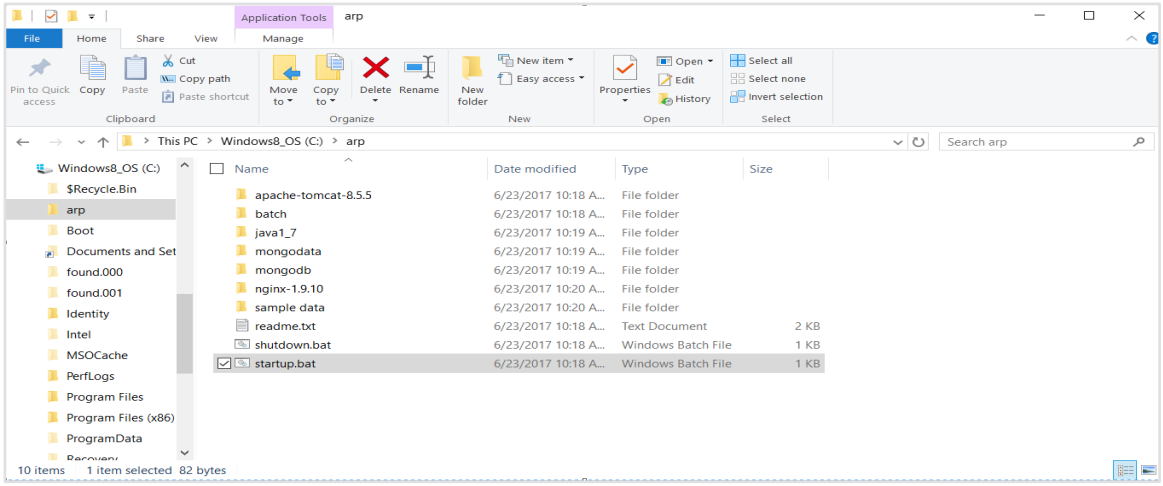
2. *Sample Data* – We have provided sample data sets for the New York network. These data sets can be accessed and uploaded into the ARP from within this folder. The sample data files provided are:
    - Road Network Nodes – rpi_nodes_nyc.csv
    - Road Network Links – rpi_links_nyc.csv
    - Entry Point Nodes – rpi_entrypoints_nyc.csv
    - Terminal Nodes – rpi_terminalpoints_nyc.csv
    - Restoration Times – rpi_restoration_nyc.csv
    - Debris Shape (SIP Output) – debris.geojson

3. *System Startup and Shutdown Instructions* – The *readme.txt* file contains quick bulleted points for starting and shutting down the ARP system for reference.

4. *Starting and Stopping ARP* – The startup.bat and shutdown.bat files will start and stop the ARP software on your machine.

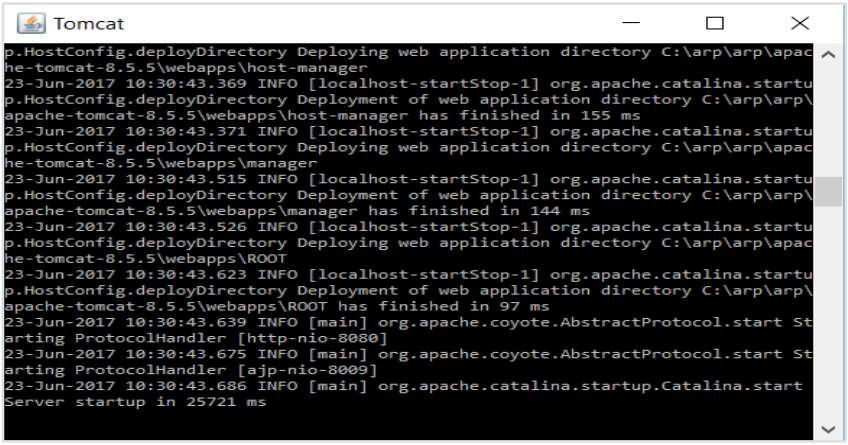
2.3 Starting the ARP Software

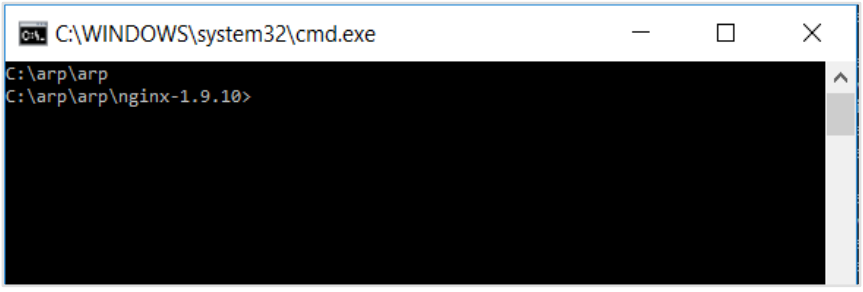
1. In order to start the ARP software on your machine, simply double click the *startup.bat* file located within the "arp" directory on your machine.

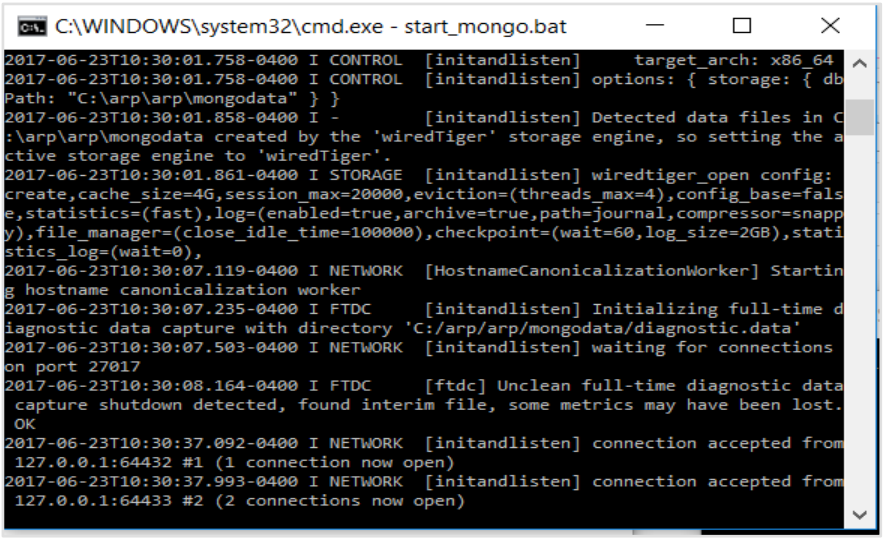


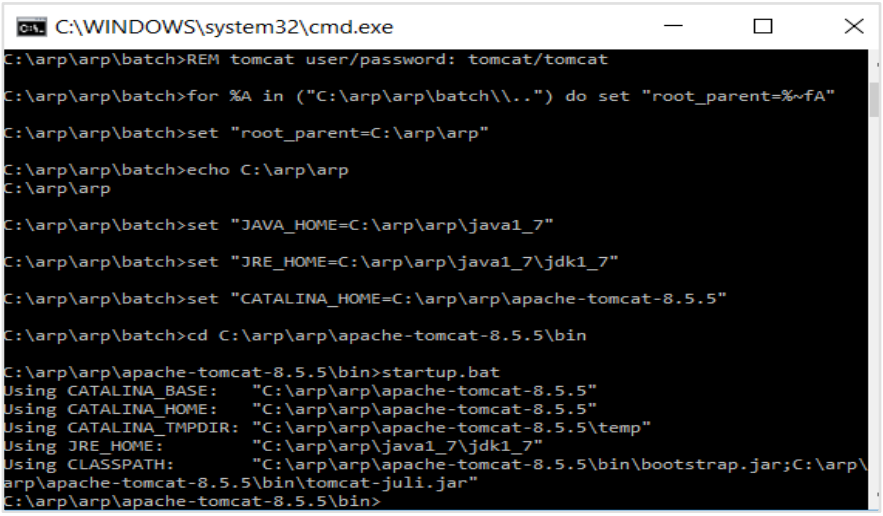
2. Once you double-click the startup.bat file you will notice four Window Command prompts appear on your machine. You can minimize (--) these prompts if you would like but DO NOT close (X) the prompts.

   NOTE: If you have a Windows firewall enabled on your machine it is possible that you will also encounter pop-ups asking if you want to allow access to your machine. Make sure that you allow access or choose proceed anyway for any firewall popup that you encounter.

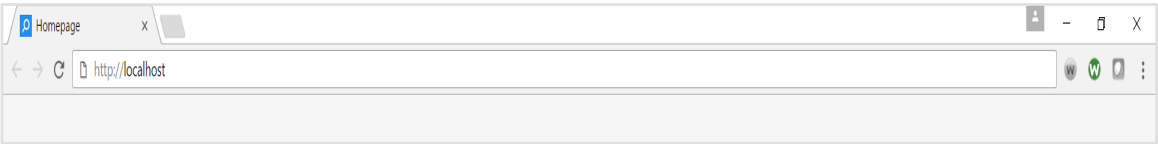




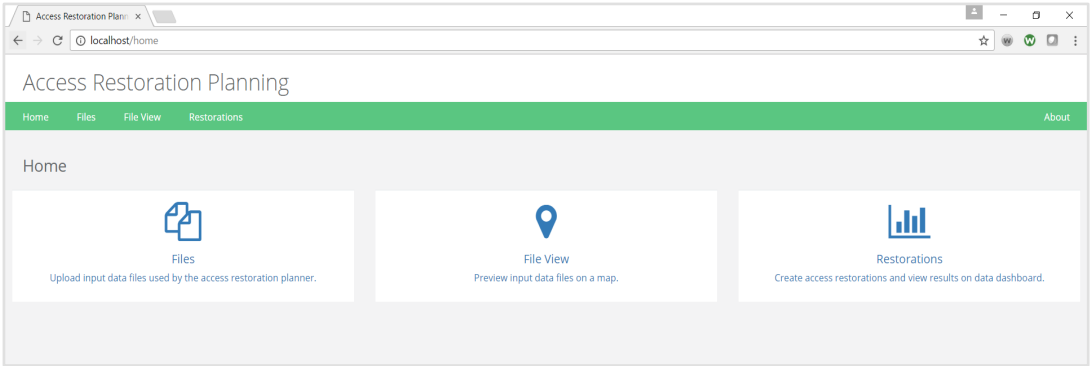




3. Open up a browser (preferably Chrome) and in the navigation bar type http://localhost

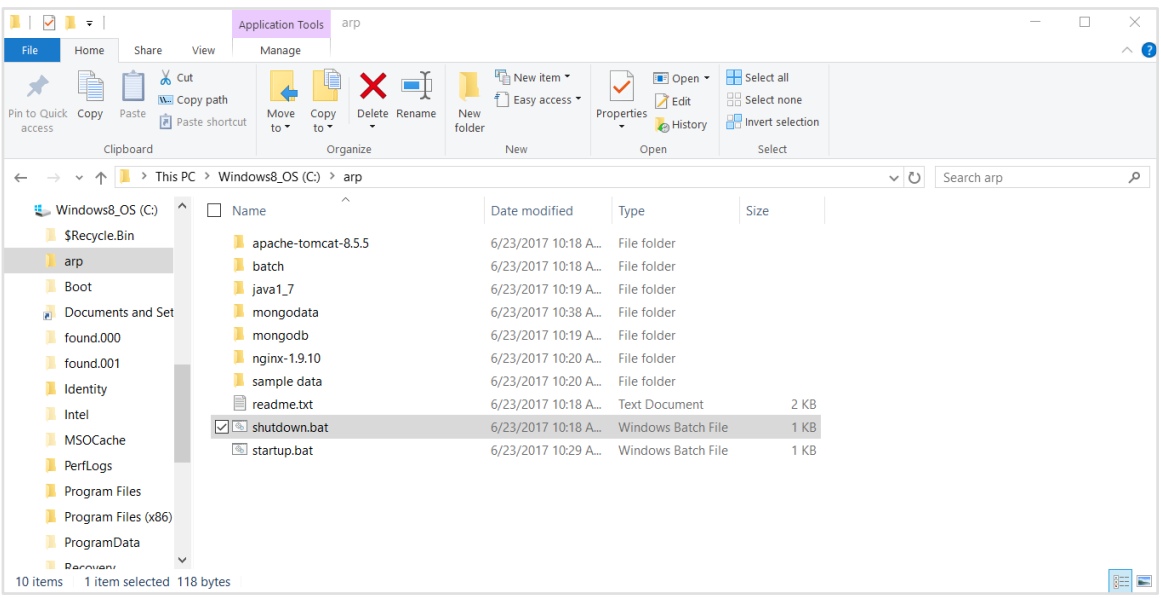


4. The ARP software will load and you will now have the ability to upload files, create/generate restorations, and view results.



2.4 Stopping the ARP Software

In order to stop the ARP software, simply double click the *shutdown.bat* file within the "arp" directory. The Windows Command prompts that previously opened will automatically close.

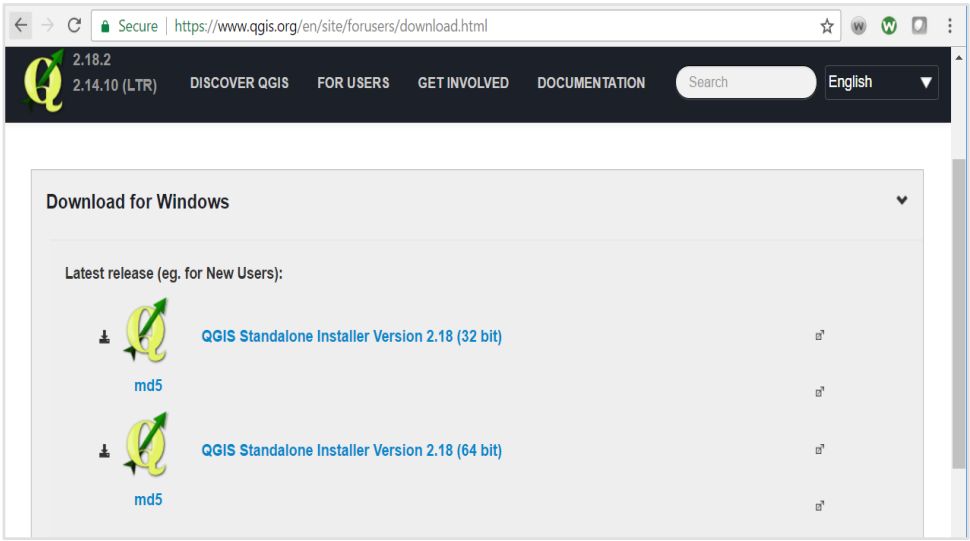


3. Geospatial Debris Data – Converting Shapefile to GeoJSON format

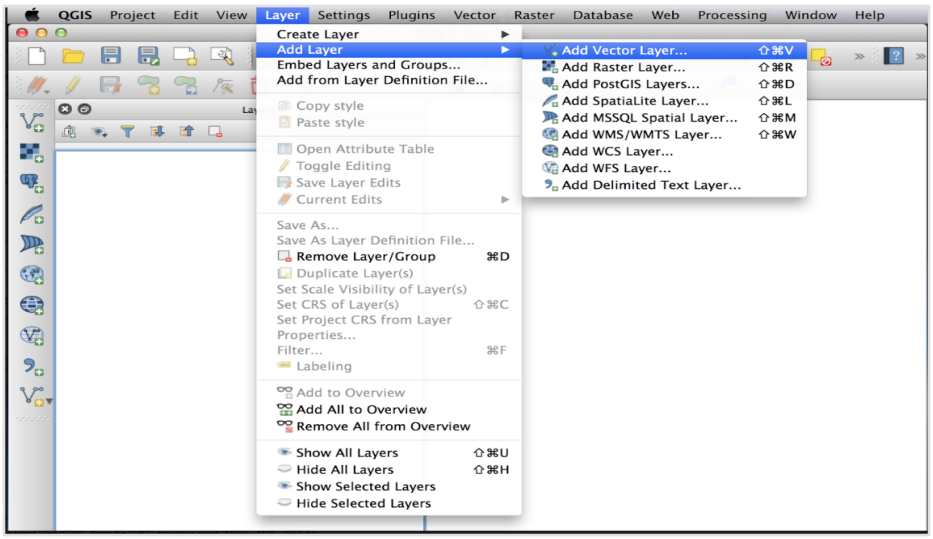
The ARP software allows users to upload, display, and process geospatial debris data in a GeoJSON format. Where necessary, users can convert Shapefiles to GeoJSON using a free, open-source GIS tool named "QGIS" using the steps below:

1. Download QGIS (which is an open source GIS data tool) from https://www.qgis.org/en/site/forusers/download.html. You can download

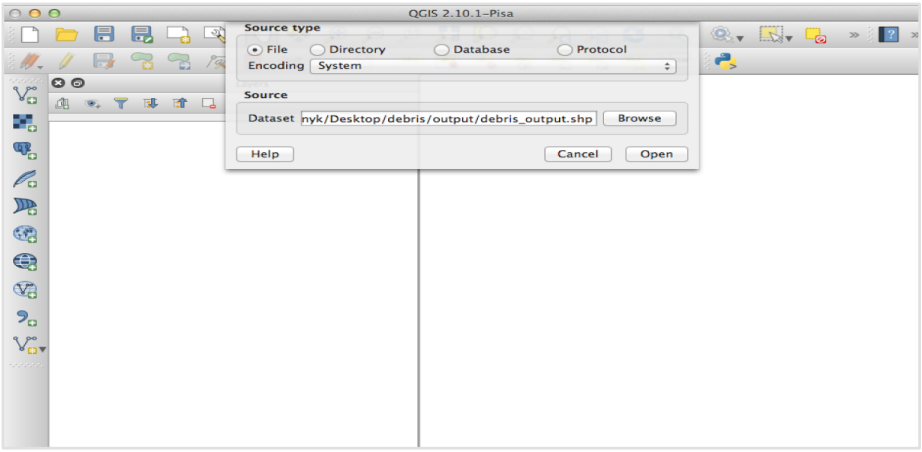
either the 32 or 64 bit version under the **Latest release (eg. for New Users) section.**



2. Once the software is downloaded and installed, launch the software. In the Header bar under the "Layer" dropdown, select "Add Layer", and then select "Add Vector Layer"

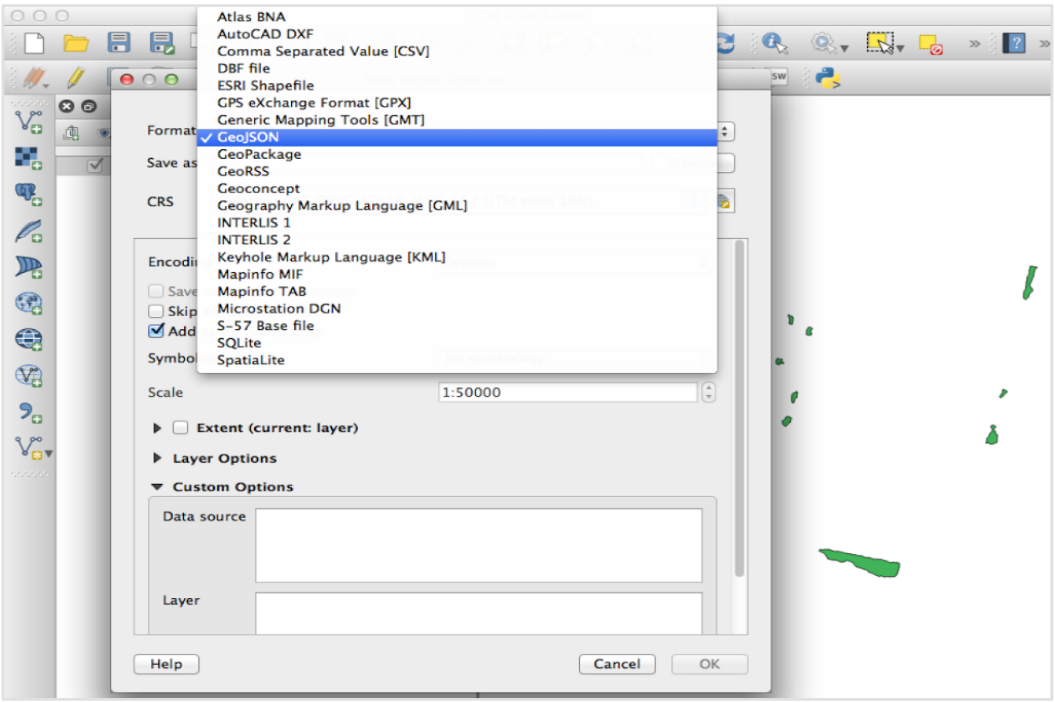


3. Under "Source" browse for and select the Shapefile (.shp file)



4. Once the shapefile data appears on the left hand side under the "Layers" area, right click on the layer name and select "Save As"

5. From the "Format" dropdown select "**GeoJSON**". From the "CRS" dropdown select "EPSG: 4326 – WGS 84 "



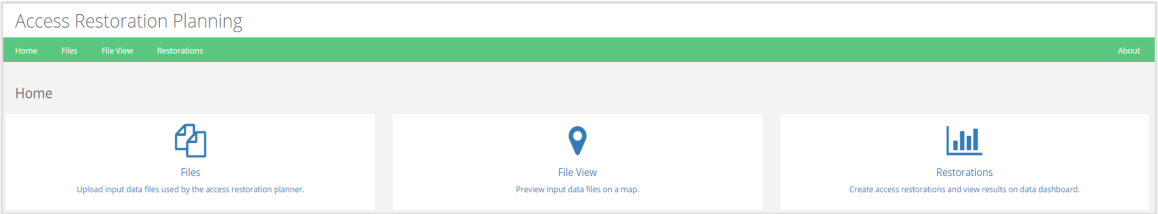
6. Click "Browse" to name and save the GeoJSON file to a specified location

7. The GeoJSON file can now be uploaded and used in the ARP application

4. Interface Workflow

4.1 Home Screen - Overview

The Access Restoration Planning Home screen is the landing page for the system. This screen provides a means for users to navigate through the software workflow.



Users can navigate to the various components of the system by either selecting the links (Home, Files, File View, and Restorations) in the header bar which is available on each screen throughout the system, or by selecting the icons:

- **Files** – Loads the module where users can upload the road network nodes, road network links, entry points, terminal points, restoration times, and debris data that are needed to generate a restoration run. Additionally, users have the ability to create entry points and terminal points using an interactive map via the *Data File Creation* screen.

- **File View** – Loads the module that allows users to review the data files for accuracy and completeness (on an interactive mapping interface) prior to generating a restoration.

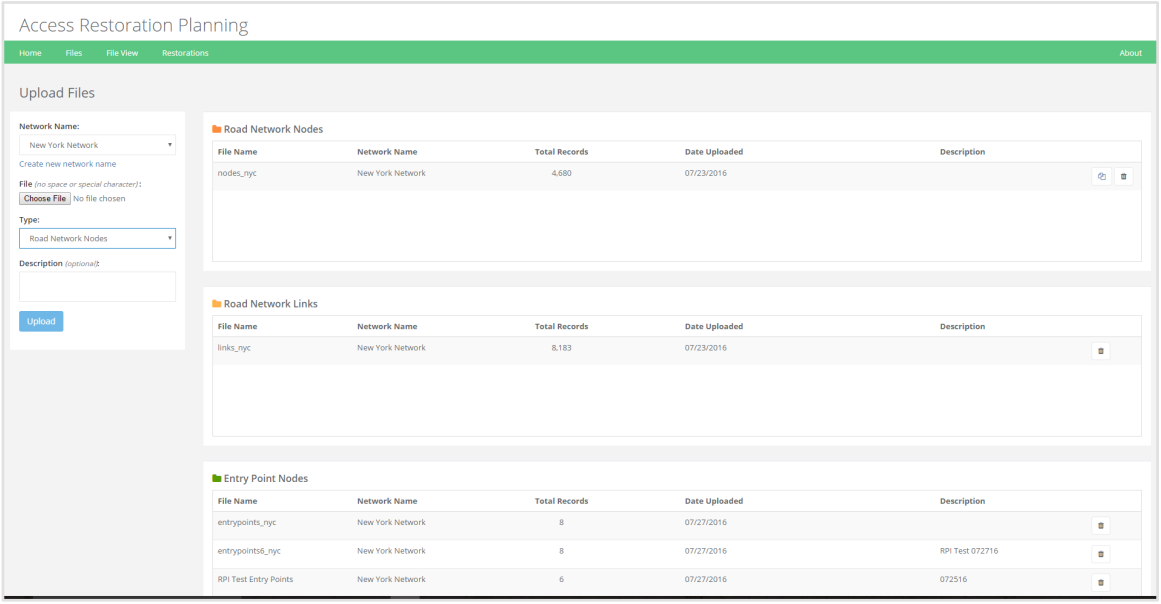
- **Restorations** – Loads the module that allows users to view a list of all restorations generated, create new restorations, delete previous restorations, and view the results of completed restoration runs.

4.2 Files (Data File Upload & Creation)

The Data File Upload screen provides users with the ability to upload the road network nodes, road network links, entry points, terminal points, restoration times, and debris data required by the restoration optimizer.

IMPORTANT:
- Files must be in a CSV format, with header and must contain all required fields (required fields defined in)
- Geospatial debris data must be in a GeoJSON format and must contain all required fields (required fields defined in)
- For a new network, <u>Node data must be uploaded first</u>. <u>Link data must be uploaded second</u>. Entry points, Terminal points, and restorations can then be uploaded (and

will be validated against the node and link data for that Network to make sure all nodes and links are accounted for).



In order to successfully upload a file you must provide the following information and select the ***Upload*** button:

- **Network Name** – This field contains a drop down of any previously created network (i.e., name of city or geographic area). The software has the ability to store multiple networks. In order to create a new network, select the "***Create New Network Name***" text.
- **Choose File** – Select the location of the file on your computer
- **File Type** – This identifies the type of file that is being uploaded. Available file types are Road Network Nodes, Road Network Links, Entry Point Nodes, Terminal Nodes, Restoration Times, and Debris Shapes
- **Description** – Optionally you can provide a description of the file that is being uploaded.

Each file that is uploaded is validated for accuracy. Users will be receive a pop-up message in both (1) the case of a successful file upload, and (2) data validation failure. Data validations performed by the software include:
- **File Name** – File name must not contain spaces or special characters
- **Entry Point and Terminal Nodes** – Node IDs must appear in the *Road Network Nodes* file for that network.
- **Restoration Times** - Link IDs must appear in the *Road Network Links* file for that network.

Each successfully uploaded file is automatically associated with the Network Name. This will allow the correct files to be available for selection on the *Create Restoration* screen. Once the file is uploaded, it will appear in the associated table on the "Files" page. The following information will also be displayed:

- **File Name**- Name of the file that was uploaded
- **Network Name** – Name of the network the file was associated with
- **Total Record** – Number of records included in the file
- **Date Uploaded** – The date/time the file was uploaded
- **Description –** A description of the file if provided

Other Functions:

- Deletes data file

- Will load the *Data File Creation* screen - allowing the user to create new Entry Node and Terminal Node files using an interactive map

4.2.1 Required Columns for Data Files

This section provides a description and required columns for each file type. Columns names must exactly match those defined below.

nodes.csv
***Description*:** Provides the unique identifier, latitude, and longitude of each node in the road network
***Required columns*:** node_id, latitude, longitude

Note: The "node_id" values provided in the nodes.csv file must be sequential integer values starting with "1". For example: 1, 2, 3, 4, 5,..., 100.

links.csv
***Description*:** Provides the unique identifier as well as the start & end node comprising each link (i.e. road segment) in the road network, along with the name of each link
***Required columns*:** link_id, node_id_1, node_id_2, link_name **

restorations.csv
***Description*:** Provides the restoration time (in hours) for each link in the road network
***Required columns*:** link_id, restoration_time

entrypoints.csv
***Description*:** Provides the listing of entry point nodes in the network
***Required columns*:** node_id, entrypoint_value **

<u>terminalpoints.csv</u>
***Description*:** Provides the listing of terminal point nodes in the network, along with their "value" (i.e. population or other), type (i.e. hospital, school, etc.), and name
***Required columns*:** node_id, terminal_value, terminal_type **, terminal_name **

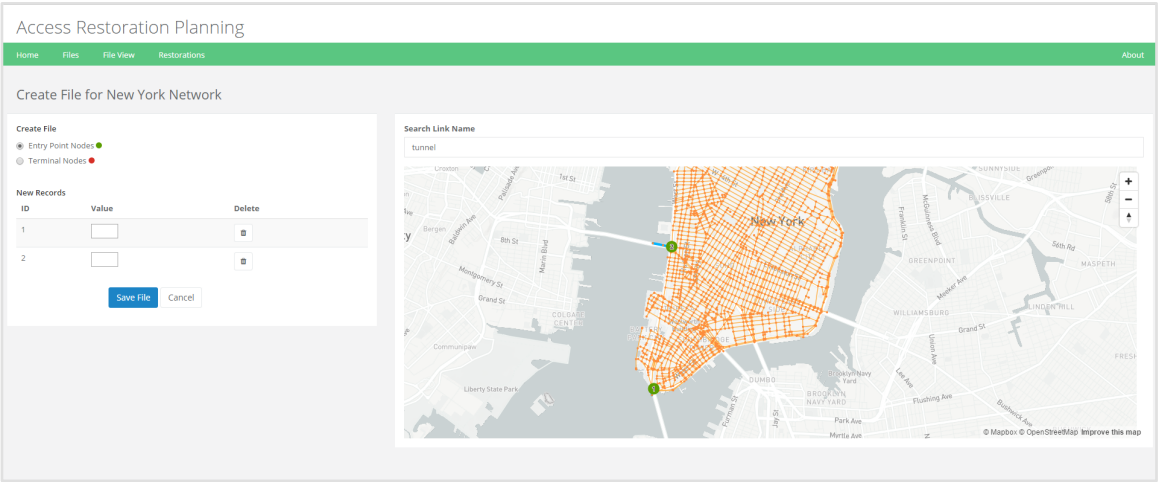
<u>debris.geojson</u>
***Description*:** Provides spatial data for debris affecting roadway passage in the network.
***Required columns*:** ID, PassWidth, Volume, (additional columns may also be included, but will be ignored)
Definitions: ID (link ID that the debris is obstructing); PassWidth (amount of passable with that is available around the debris, in meters); Volume (volume of debris, in cubic meters)

** column must exist in the file, but can be left blank

Note: To get a better understanding of required columns and data for each file type, users can download sample files for NYC on the "About" page.

4.2.2 Entry Point and Terminal Point File Creation

In the "Road Network Nodes" table, click on the icon (on the right side of the table) for any <u>node dataset</u>. The *Data File Creation* screen presents users with a geographic view of the road network node and link data files. Users can search for various keywords in the *link_name* column (from links.csv file) and optionally create new Entry Point Node and Terminal Node files that can be used into the restoration optimizer. When the page loads the user will be presented with a representation of all network node for the network selected.



Creating a new Entry Point File

Select (1) the Entry Point Node radio button, and then (2) click on the nodes on the map. A search box is also provided to help identify and narrow down nodes of interest. For example, enter "tunnel" into the "Search Link Name" text box highlight links with the word "tunnel" in blue on the map (such as Lincoln Tunnel or Holland tunnel). You can then select entry point nodes (represented by green dot) associated with those links on the map and a record will be created for each node selected. Save the file, by selecting the ***Save File*** button. Enter a name for the file you are about to create and the file will be available under the Entry Point Node section on the *Data File Upload* screen.

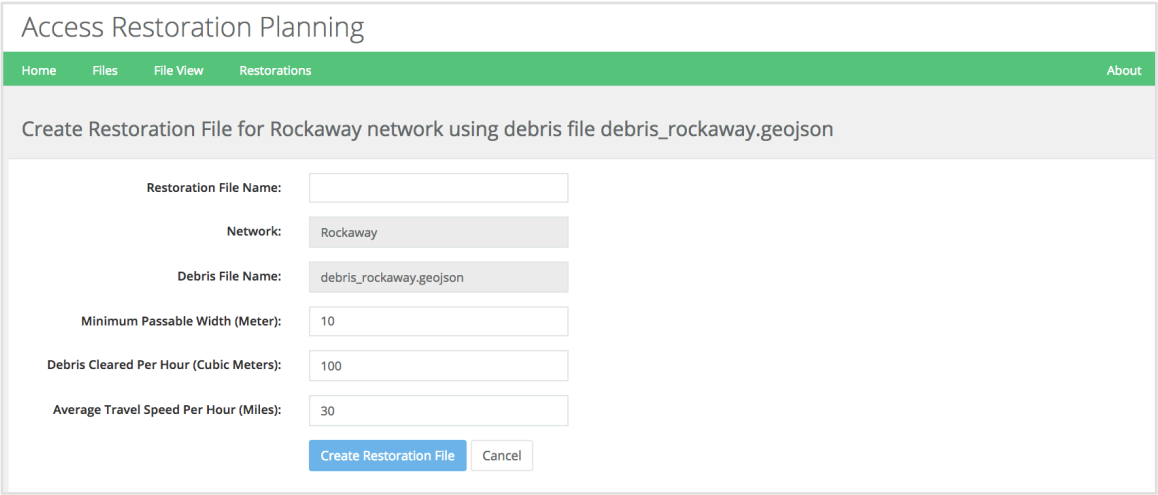
Creating a new Terminal Point File

Use the same process to create a Terminal Node file (comprised of hospitals, schools, etc.). Terminal nodes will be represented by a red dot on the map.

4.2.3 Using debris data to calculate restoration times

In the "Debris shapes (SIP output)" table, click on the icon (on the right side of the table) for any debris dataset. This functionality allows users to generate restoration times for links in a network based on the debris in the road network. The user will be prompted for the following information:

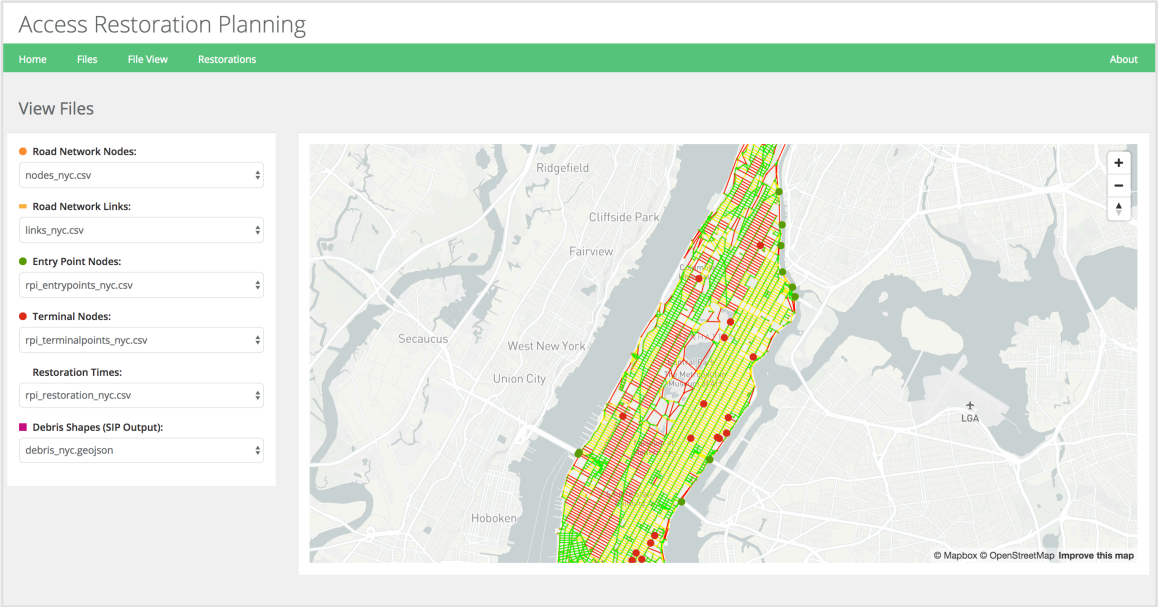
- **Restoration File Name:** Name of the restoration file that the user will create
- **Minimum Passable Width (meters):** Defines the minimum passable width required for each link (i.e. roadway). Debris obstructing the roadway (not allowing for this passable width) will be cleared
- **Debris Cleared Per Hour (cubic meters):** Amount of debris volume that equipment can clear in one hour
- **Average Travel Speed (miles per hour):** Average travel speed of equipment traveling on this road network



To run the restoration time calculation and save the file, click the "Create Restoration File" button. The newly-created file will appear in the "Restoration Times "table on the "Files" page. In addition, a CSV file of the result will automatically be downloaded to the default download location on your machine.

4.3 File View - Data File Visualization

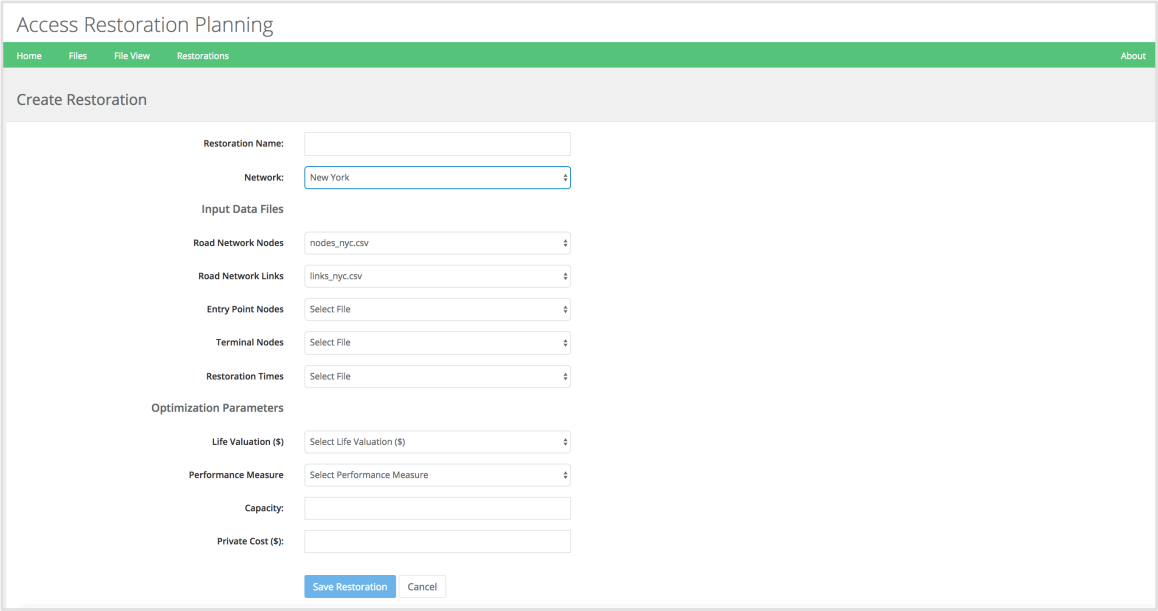
The File View screen allows a user to view uploaded data files on a map. For a specified road network (i.e. city or geographical area), the user can visualize nodes, links, restoration times, entry points, terminal nodes, and debris associated with the network. In order to view this information, simply select files from the drop down list associated with each data file type. The map will automatically zoom to the extent of the data.



4.4 Restorations – Create and Run Restorations

4.4.1 Create Restoration

To create a new Restoration scenario, click the "Create Restoration" button. The Create Restoration page provides a platform by which a user can provide all information (i.e., data files and parameters) necessary to create and run an optimal restoration plan.



Restoration Name and Network

In order to create the restoration you must provide a unique name for the restoration and select the network on which to create an optimal restoration.

Input Data Files

After selecting the network, parameter selections will appear on the page. Under "Input Data Files", drop downs will pre-populate for the 5 different file based on the network selected. Select the file you wish to use for the restoration run.

Optimization Parameters

The user must then select various optimization parameters including:

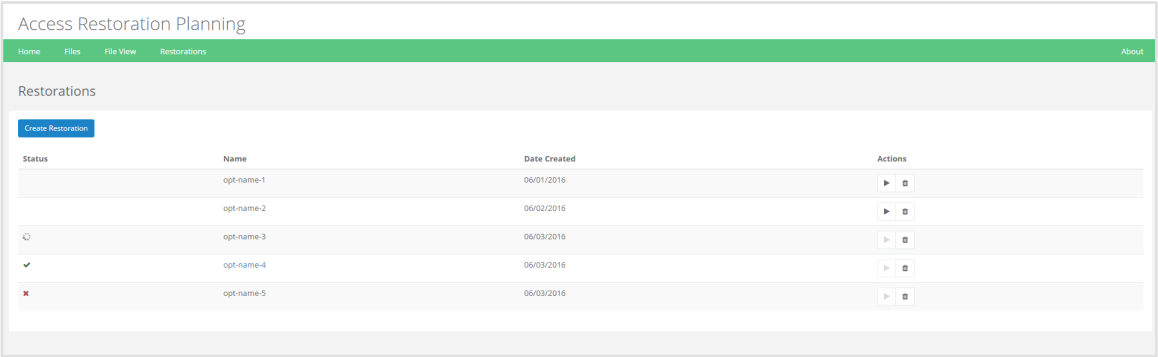
- **Life Valuation –** Value of human life in dollars
- **Performance Measure –** Type of performance measure to use
- **Optimization Routine –** Metric on which to optimize (only appears when selecting "Small SC, POP, SC, MC_OP and MC_SC" Value in "Performance Measure" selection
- **Capacity –** Equipment quantity available to clear network (note: only one piece of equipment may clear 1 link at one time)
- **Private Cost –** Private cost associated with the network

Once all the Input Data Files and Optimization Parameters have been provided select the ***Save Restoration*** button to add the restoration to the *Restoration* screen.

4.4.2 Run Restoration

The main *Restoration* screen provides a listing of each restoration that has been created. For each restoration listed the following details are provided: 1) the status

of the restoration, 2) the name provided for the specific restoration, 3) the date the restoration was created, and 4) the ability to either run or delete the restoration.

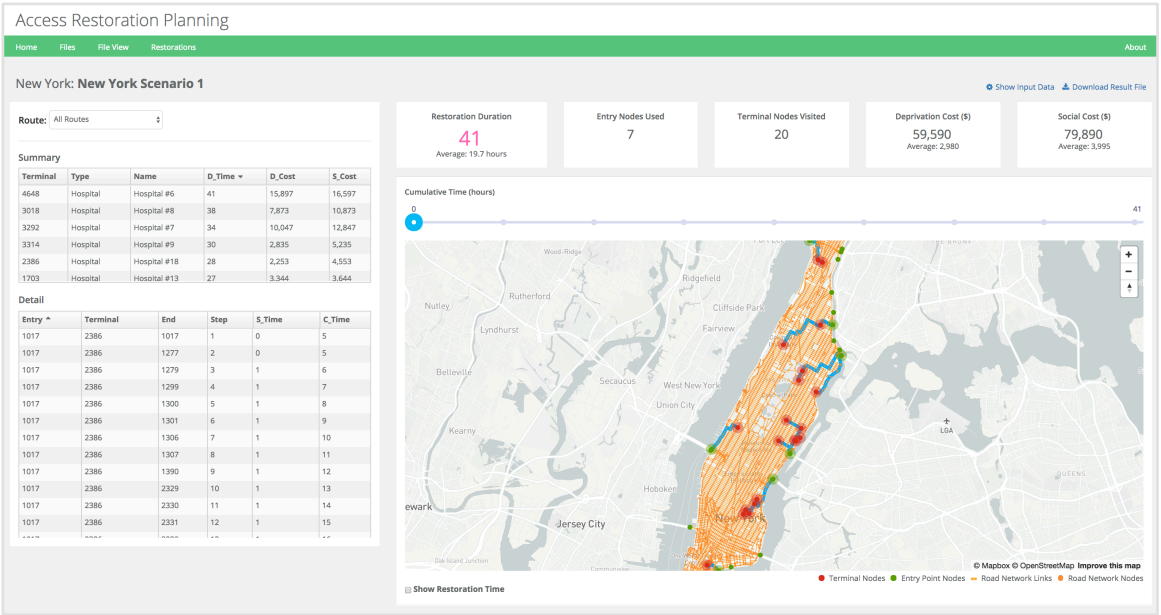


Each restoration listed on the screen will have a status associated. If no status is associated with the restoration then the restoration has been created but has not yet been processed. A restoration that has a icon means that the restoration has successfully completed and results are available for that restoration. If an error occurs during the processing the user will be notified via error message.

In order to process, or run, a restoration select the icon in the action column associated with the restoration. Each restoration can only be processed once. If you would like to delete a restoration from the history select the icon.

4.4.2 Restoration Results and Visualization

The *Restoration Results and Visualization* screen provides the results from the restoration optimization run. The results provide an overall summary of the restoration run as well as data tables that provide information regarding the restoration time for each entry point and terminal node combination. In addition to the overall summary and data tables, users can visualize the optimal restoration route from entry point to terminal point, as well as the time for restoration to complete on a map.



The user has the ability to view the data files and input parameters for this restoration optimization run by selecting the "*Show Input Data*" link. This data can be removed from the screen by selecting the "*Hide Input Data*" link. Restoration results can be downloaded by clicking the "*Download Result File*" link.

The overall summary provides information regarding:

- **Restoration Duration -** The total restoration time (i.e. time it took to reach all terminal nodes) in hours
- **Entry Points Used -** The number of distinct entry nodes used by equipment to reach all terminal nodes
- **Terminal Nodes Visited** - The number of terminal nodes reached in the restoration run
- **Deprivation Cost** - The total deprivation cost associated with the restoration in dollars
- **Social Cost** - The total social cost associated with the restoration in dollars

On the left hand side, two tables provide more detailed information regarding the optimal restoration solution. These tables provide a summary of (1) restoration time, deprivation cost, and social cost for each terminal nodes, and (2) the step-by-step path taken to reach each terminal node n the solution.

The summary table provides the following information:

- **Terminal ID –** ID associated with terminal node
- **Type** – The type of terminal point, i.e., school, government, hospital
- **Name** – The name of each terminal point
- **D_Time** – Deprivation time for each terminal node
- **D_Cost** – The deprivation cost for each terminal node
- **S_cost** – The social cost for each terminal node

Finally, the map representation depicts the optimal restoration paths determined by the model. The entry points are depicted by a “green dot”, while the terminal points are depicted by a “red dot”, and the restoration path is indicated by a “blue line”. The user can hover over the nodes and links to view additional information.

The map includes a time-lapse scroll bar. As the user scrolls the bar to the right, the map changes to show the restoration progress up until that time. This is visualized using a changing color in the restoration route. The “blue line” changes to a “green line” to indicate completion.

APPENDIX G:
Debris Utility User Manual

DebrisUtility v1.0 User Manual

September 27, 2016

Contents

1 Overview **2**

2 Installation **2**

3 Running the Application **6**

4 The Main Menu Interface **6**
- 4.1 Single File Mode . . . 7
- 4.2 Batch File Mode . . . 7

5 The Single File Mode Interface **8**
- 5.1 The Toolbar . . . 8
  - 5.1.1 Zoom In . . . 9
  - 5.1.2 Zoom Out . . . 9
  - 5.1.3 Pan . . . 9
  - 5.1.4 3D Rotation . . . 9
- 5.2 The Input/Output File Panel . . . 10
  - 5.2.1 The LAS Section . . . 10
  - 5.2.2 The SHP Section . . . 10
  - 5.2.3 The Output Directory Section . . . 11
  - 5.2.4 The Back Button . . . 11
  - 5.2.5 The Text Display . . . 12
- 5.3 The Data Visualization Panel . . . 12
  - 5.3.1 The Inputs Dropdown Menu . . . 12
  - 5.3.2 The Outputs Dropdown Menu . . . 14
- 5.4 The Run Settings Panel . . . 15
  - 5.4.1 The Units Panel . . . 16
  - 5.4.2 The Algorithm Parameters . . . 16

5.4.3 The Restore Defaults Button 17
5.4.4 The Run Button 17
5.5 Running an Example 18

6 The Batch File Mode Interface **21**
6.1 The Input/Output File Panel 21
6.1.1 The Input Directory Section 22
6.1.2 The Output Directory Section 22
6.1.3 The Back Button 22
6.1.4 The Text Display 23
6.2 The Run Settings Panel 23
6.2.1 The Units Panel 23
6.2.2 The Algorithm Parameters 24
6.2.3 The Restore Defaults Button 24
6.2.4 The Run Button 25
6.3 Running an Example 25

1 Overview

This is the user manual for the DebrisUtility v1.0 application. The application provides a user interface for generating roadway debris maps using commercial remote sensing (CRS) data. The underlying algorithms take an input point cloud LAS file (*.las) and a corresponding road network shapefile (*.shp), detect debris piles, estimate their volumes, and then save a debris and road shapefile. The manual covers everything from installation to outputs and describes the function of every feature and component of the application. Screenshots are included with the descriptions as well as a start-to-finish example so that the user can follow along and get a feel for flow of the application.

2 Installation

This section takes the user step-by-step through the installation process.

1. Begin by running the MyAppInstaller_web application. This will open the DebrisUtility Installer, a step-by-step application that will guide you through the installation process. Note that an internet connection is required to install the application. If your internet connection requires a proxy server (this is rarely the case), configure that by hitting the "Connection Settings" button. Otherwise, hit the next button.

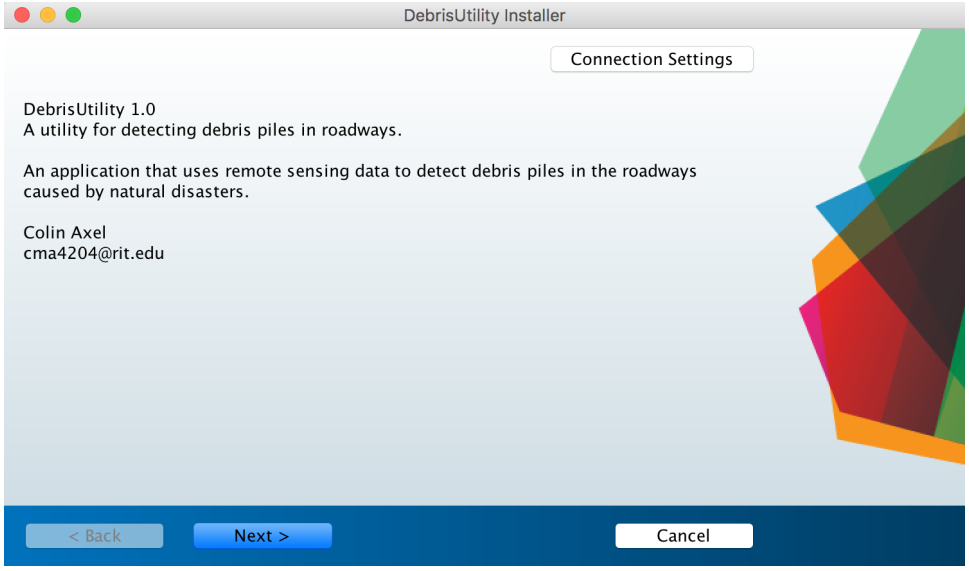


Figure 1: The DebrisUtility Installer window

2. The next window that will pop up is the Installation Options window. Choose an installation directory using the browse button, and then hit next.

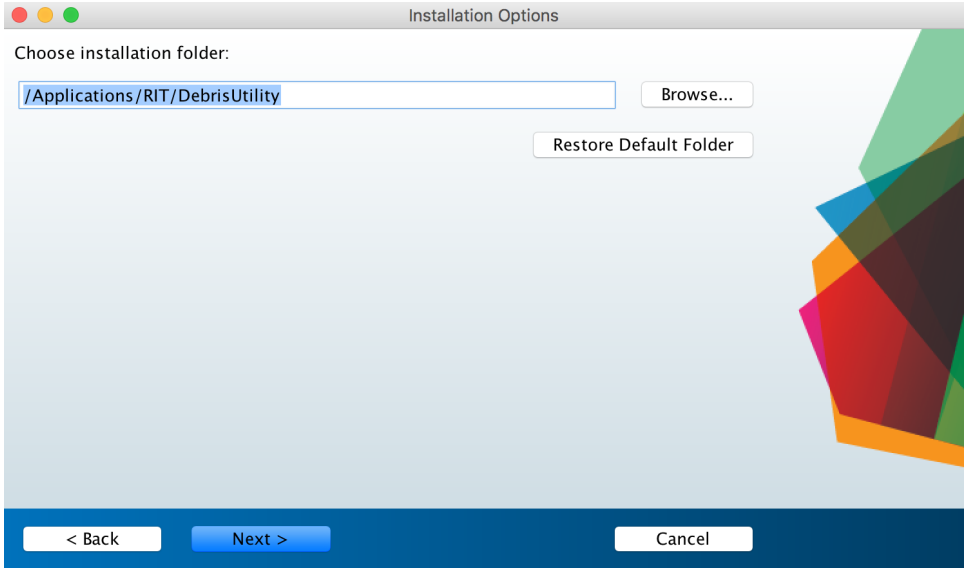


Figure 2: The Installation Options window

3. The Required Software window will inform you that the MATLAB Runtime software is required. The application cannot run without this piece of software, so install it either in the default directory or another directory of your choice. If MATLAB Runtime is already installed on your computer it will automatically be detected. Hit the next button.

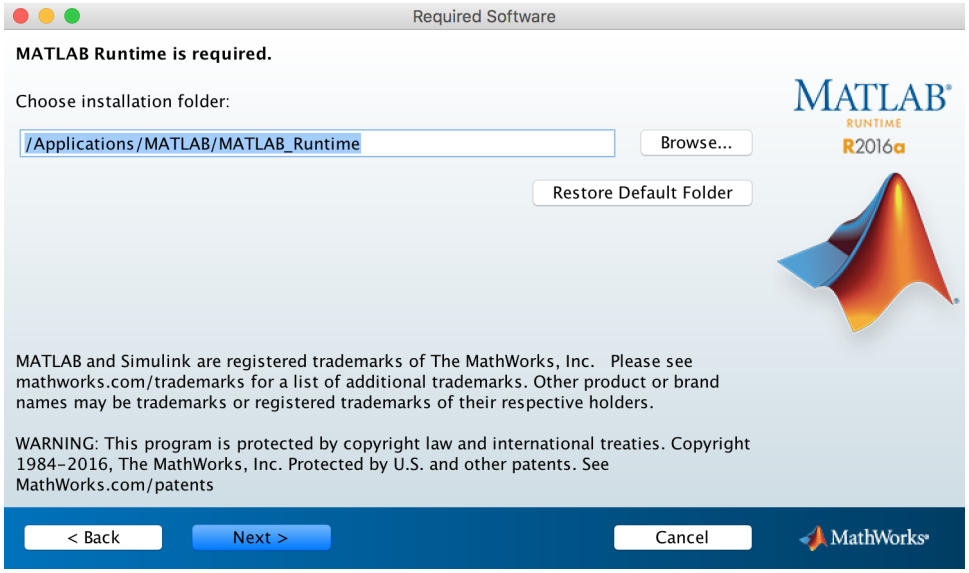


Figure 3: The Installation Options window

4. The Confirmation window will pop up and provide some basic information about the MATLAB Runtime and where it will be installed. Hit the Install > button.

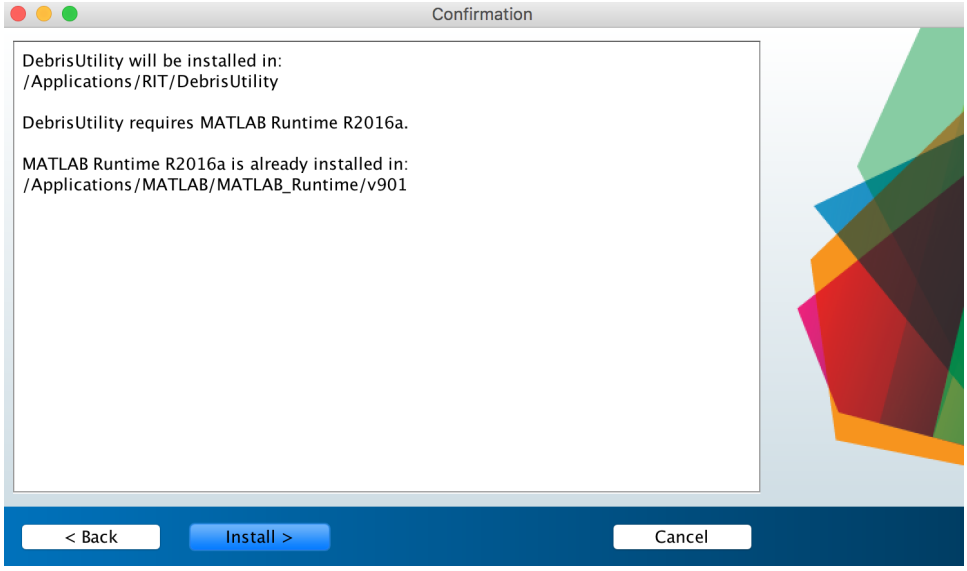


Figure 4: The Confirmation window

5. The installer will take a few moments to install and then the Installation Complete window will appear, letting you know that the installation completed successfully. Hit the finish button. At this point the application is installed and ready to use.

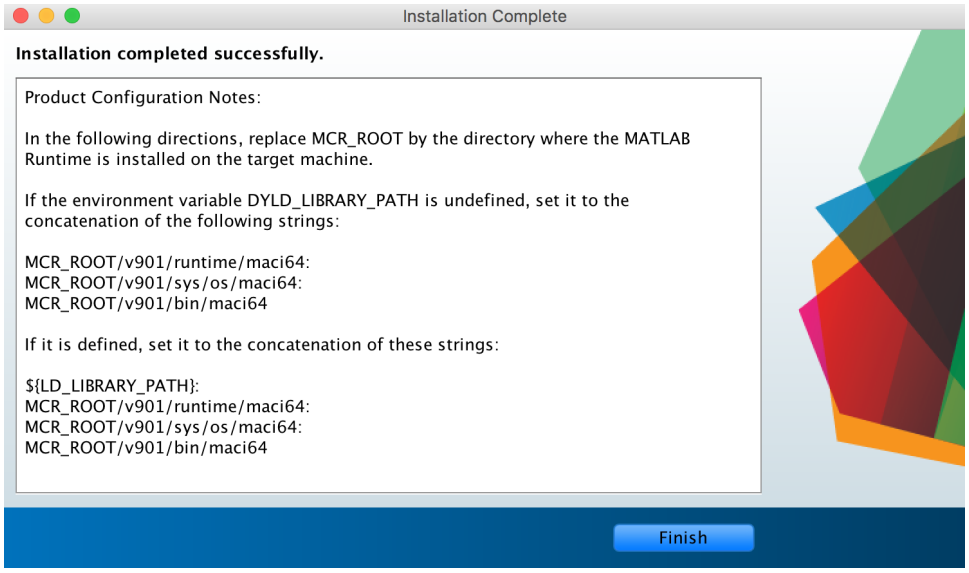


Figure 5: The Installation Complete window

3 Running the Application

The location of the application will depend on the directory that was selected during installation. For example, the path on this machine is: "/Applications/RIT/DebrisUtility". Navigate to that directory and location the "application" folder. Inside of this folder you will see the DebrisUtility application. Double click the application to run it.

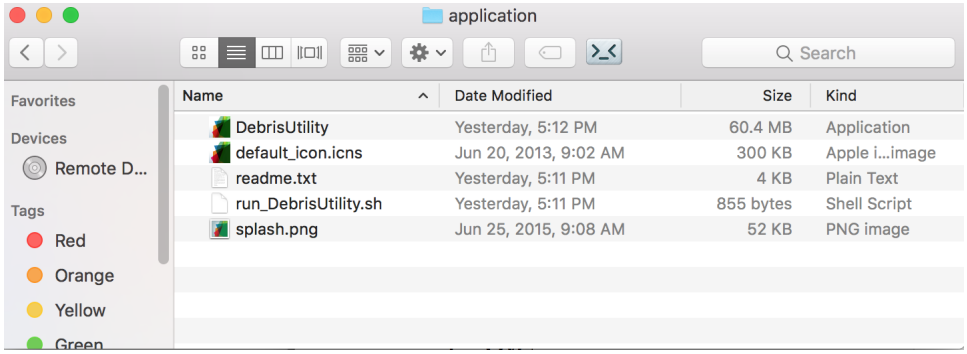


Figure 6: Locating the DebrisUtility application

4 The Main Menu Interface

The first window that pops up when running the application is the main menu. The main menu is a simple interface that presents the user with two run mode options: Single File Mode and Batch File Mode.

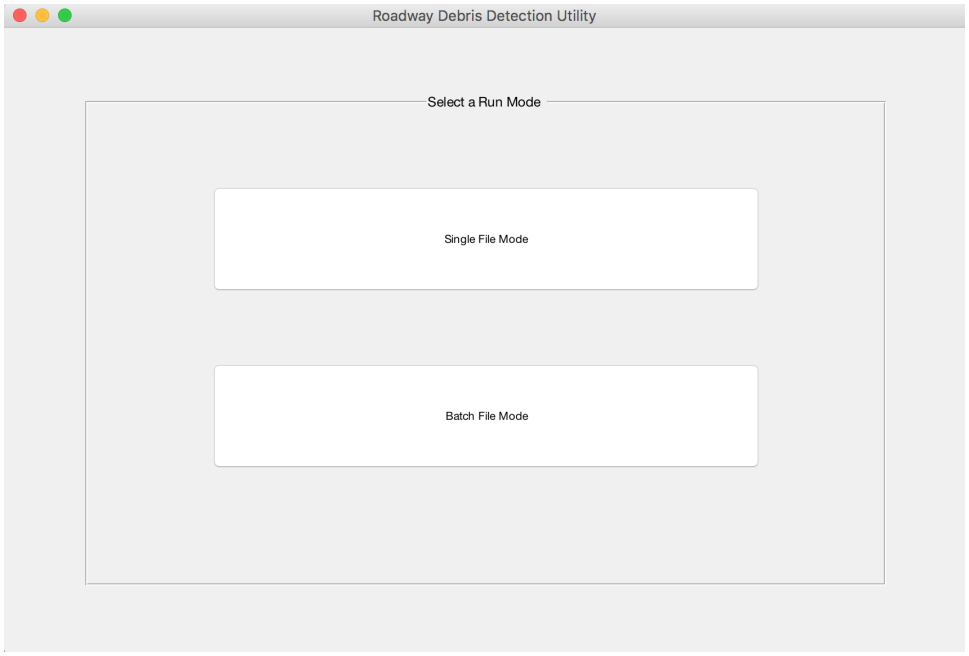


Figure 7: The main menu interface

4.1 Single File Mode

Single file mode is used to perform roadway debris detection and volume estimation on a single point cloud and requires a single LAS (*.las) and road network shapefile (*.shp). Single file mode requires the user to manually load one file at a time, adjust settings (optional), and then run the algorithm. This mode is best to use when you want to be able to manually adjust settings and view the results for a single file at a time.

4.2 Batch File Mode

Batch file mode is used to perform roadway debris detection and volume estimation on all point clouds in a selected directory and requires one or more LAS (*.las) and road network shapefiles (*.shp) with the same names. Batch file mode requires the user to select an input directory (that contains the files to run), adjust settings (optional), and then run the algorithm for all files in the input directory. This mode is best to use when you want to let the program run for several files without needing any user interaction other than the initial setup.

5 The Single File Mode Interface

The single file mode interface pops up when selecting single file mode from the main menu. This interface contains all of the controls and features needed to perform debris detection, volume estimation, and shapefile output on a single point cloud. The interface is divided into four sections: (1) the toolbar, (2) the input/output file panel, (3) the data visualization panel, and (4) the run settings panel. The interface with the four sections labeled is displayed in Figure 8. Each of these four sections and their components will be described in separate subsections.

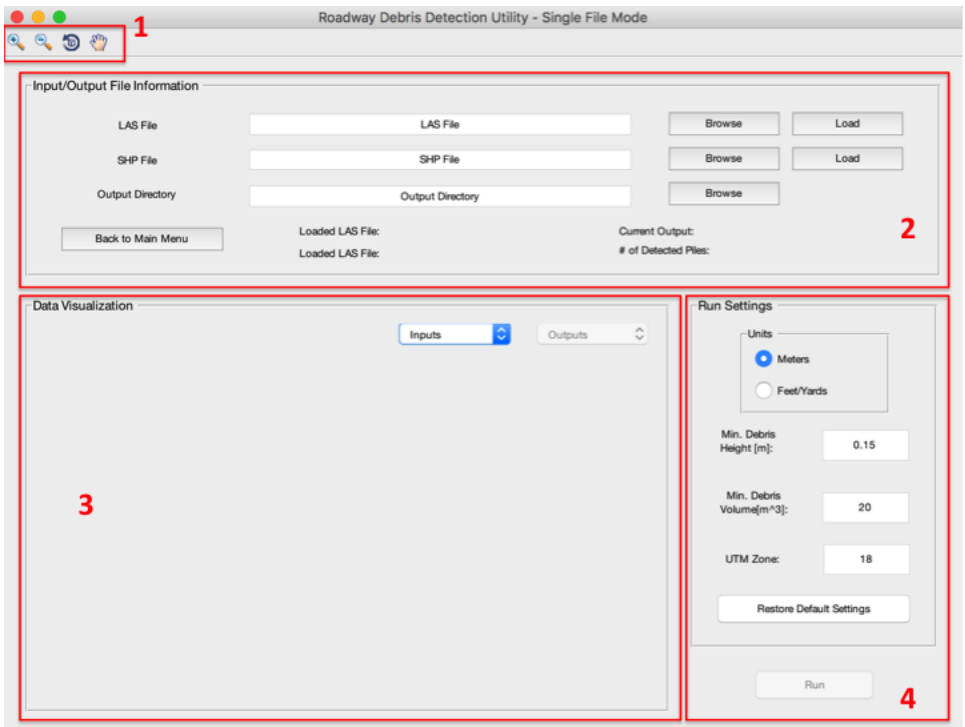


Figure 8: The single file mode interface

5.1 The Toolbar

The toolbar (the box labeled 1 in Figure 8) contains four tools thats can be used when viewing point clouds, road networks, or combined point clouds/road networks in the data visualization panel. Different input and outputs can be visualized by selecting one of the options from the input or output menu. Once a feature has appeared in the visualization

panel, the four icons on the toolbar let you do four things: zoom in, zoom out, pan, and rotate in 3d.

(a) (b) (c) (d)

Figure 9: The four tools of the toolbar: (a) zoom in (b) zoom out (c) pan (d) 3D rotation

5.1.1 Zoom In

This tool allows the user to zoom in when viewing a point cloud, road network, or debris piles. Sometimes point clouds are very large and it is hard to see what is going on when viewing the whole cloud. In these cases, use the zoom in tool to view a specific region. When the tool is selected, click the image to zoom into a specific point or click and drag a box to zoom into a whole region.

5.1.2 Zoom Out

This tool allows the user to zoom out when viewing a point cloud, road network, or debris piles. This tool only works after the screen has already been zoomed in. When the tool is selected, click the image anywhere to zoom out.

5.1.3 Pan

This tool allows the user to navigate the display after it has been zoomed in. When the tool is selected, click the image and drag.

5.1.4 3D Rotation

This tool allows the user to rotate the display in 3D. It is most useful when viewing a 3D feature such as a point cloud, debris pile points, or debris pile alpha shapes. When the tool is selected, click the image and drag to rotate. When the tool is selected, the user can also right click to return to the original view or a pre-set rotation (such as the X-Y axis view).

5.2 The Input/Output File Panel

The input/output file panel (the box labeled 2 in Figure 8) is where the user selects and loads the input point cloud, input road network, and shapefile output directory. Additionally, the user can choose to return to the main menu from this panel. Once a LAS file, SHP file, and output directory have been selected, the "Run" button will become clickable and the algorithms can be executed. It should be noted that the LAS and SHP file must both be in the same projection (UTM). For the sake of this manual, the input/output panel can be divided into five subsections: (1) the LAS section, (2) the SHP section, (3) the Output Directory section (4), the back button, and (5) the text display.

5.2.1 The LAS Section

The LAS section contains all of the tools needed to select and load a point cloud LAS file. The text edit bar allows the user to manually enter the filename (including path and extension) of a LAS (*.las) file either by typing or copying and pasting the path. If the entered path doesn't lead to a valid LAS file, an error will appear. The browse button lets the user navigate to a LAS file, and will also prevent the user from selecting a file that isn't a valid LAS file. Once a valid file has been selected (if the filename appears in the text edit bar it means it is valid) the user can then load the file by simply hitting the load button. Once a file has been loaded it can be viewed in the data visualization panel. At any point, the user can select and load a new LAS file (note that it won't be loaded until the load button is clicked). Once a LAS file is loaded, the filename will appear in front of the "Loaded LAS File:" section of the text display.

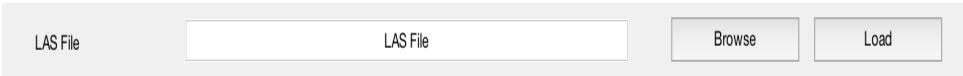


Figure 10: The LAS section

5.2.2 The SHP Section

The SHP section contains all of the tools needed to select and load a road network shapefile. The text edit bar allows the user to manually enter the filename (including path and extension) of a SHP (*.shp) file either by typing or copying and pasting the path. If the entered path doesn't lead to a valid shapefile, an error will appear. The browse button lets the user navigate to a shapefile, and will also prevent the user from selecting a file that isn't a valid shapefile. Once a valid file has been selected (if the filename appears in the text edit bar it means it is valid) the user can then load the file by simply hitting the load

button. Once a file has been loaded it can be viewed in the data visualization panel. At any point, the user can select and load a new shapefile (note that it won't be loaded until the load button is clicked). Once a shapefile is loaded, the filename will appear in front of the"Loaded SHP File:" section of the text display.

Figure 11: The SHP section

5.2.3 The Output Directory Section

The Output Directory section contains all of the tools needed to select a directory in which the output shapefiles will be saved. The text edit bar allows the user to manually enter a directory by typing or copying and pasting the path. If the entered path doesn't lead to a valid directory, an error will appear. The browse button lets the user navigate to an output directory At any point, the user can select a new output directory.

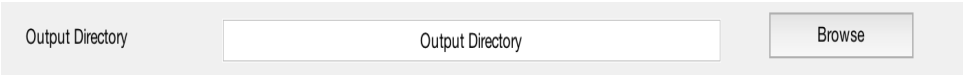


Figure 12: The Output Directory section

5.2.4 The Back Button

The back button serves one simple purpose - it brings the user back to the main menu. The back button should only be used when the user wants to switch running modes (from single to batch or vice versa). All currently loaded data will be cleared when hitting the back button (except for saved shapefiles, they will remain in the selected output directory).

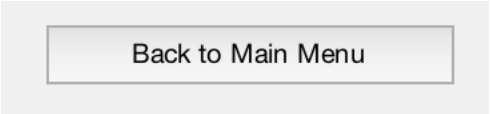


Figure 13: The Back Button

5.2.5 The Text Display

The text display exists mainly so that the user can reference what files are currently loaded into the application. The "Loaded LAS File:" display lets the user know which LAS file is currently loaded - this file is what will be used for visualization and running (note: even if a new file has been selected, until it has been loaded the old file will remain). The "Loaded SHP File:" display lets the user know which SHPfile is currently loaded - this file is what will be used for visualization and running (note: even if a new file has been selected, until it has been loaded the old file will remain). The "Current Output:" display is populated once the algorithms have completed running and lets the user know what output is currently in the system (for visualization purposes). The "# of Detected Piles:" display is populated one the algorithms have completed running and simply shows the number of debris piles detected by the algorithms.

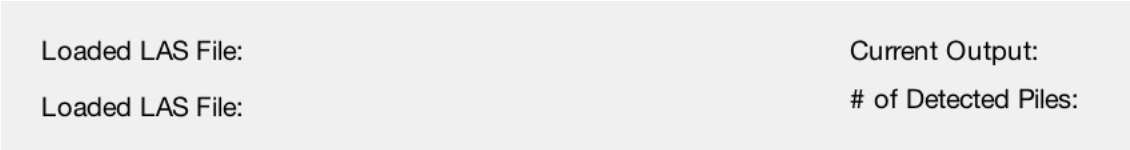


Figure 14: The Text Display section

5.3 The Data Visualization Panel

The data visualization panel (the box labeled 3 in Figure 8) is where the user can visualize the inputs and outputs (once generated). The data visualization panel doesn't become activated until the user chooses a input or output to view from the dropdown menus. The rest of this section will give examples of what the data visualization panel looks like for the various inputs/outputs.

5.3.1 The Inputs Dropdown Menu

The inputs dropdown menu lets the user choose from three different choices: point cloud, roads, or both. If the user tries to visualize a file that hasn't been loaded yet, an error message will pop up.

The point cloud selection will display a point cloud colored by elevation. The user can zoom, pan, and rotate in 3D.

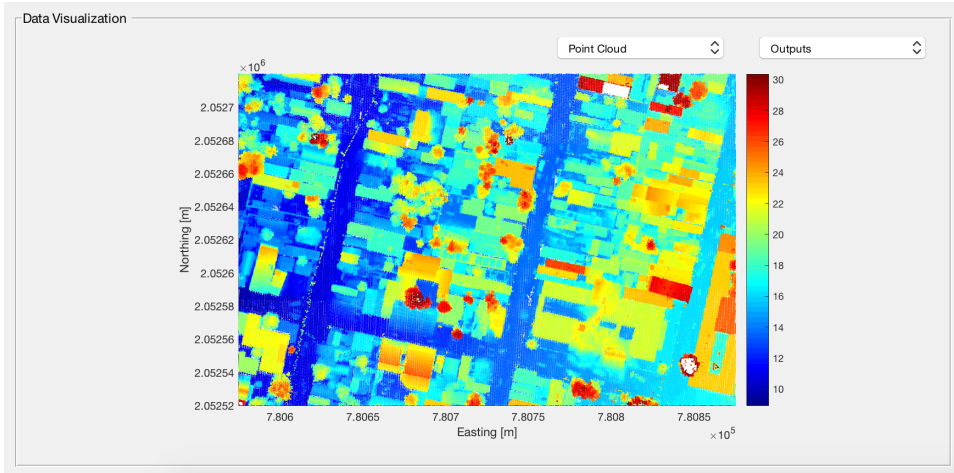


Figure 15: Visualizing a point cloud

The roads selection will plot the road network (each road is outlined in black). The user can zoom and pan.

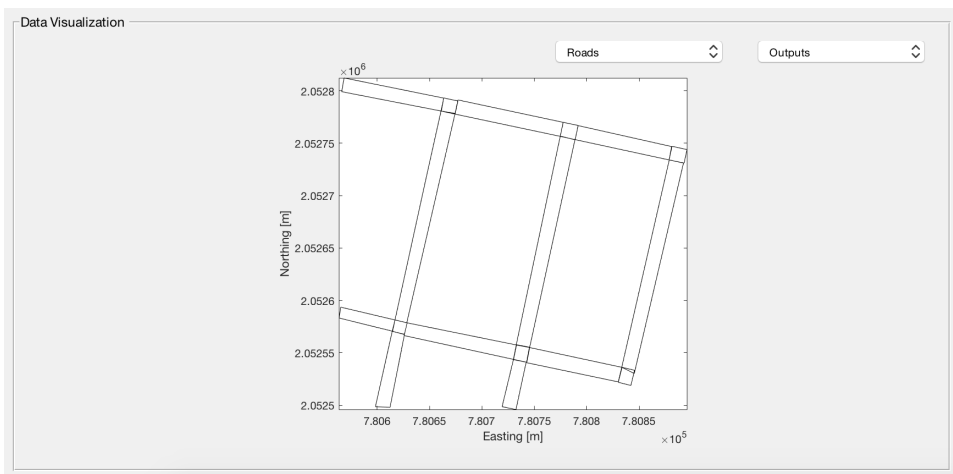


Figure 16: Visualizing the road network

The both selection will plot the road network (each road is outlined in black) on top of a 2D view of the point cloud. The user can zoom and pan.

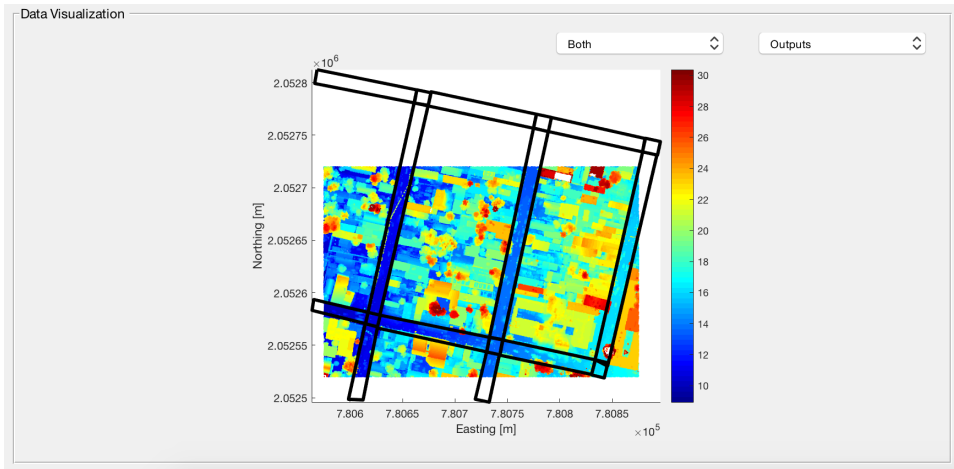


Figure 17: Visualizing both the point cloud and the road network

5.3.2 The Outputs Dropdown Menu

The outputs dropdown menu lets the user choose from two different choices: debris points and debris surfaces. The outputs menu is only accessible after the algorithms have completed and the outputs have been generated.

The debris points selection will display the individual debris pile point clouds (each unique pile has a different color) on top of the roadway points (displayed in black). The user can zoom, pan, and rotate in 3D.

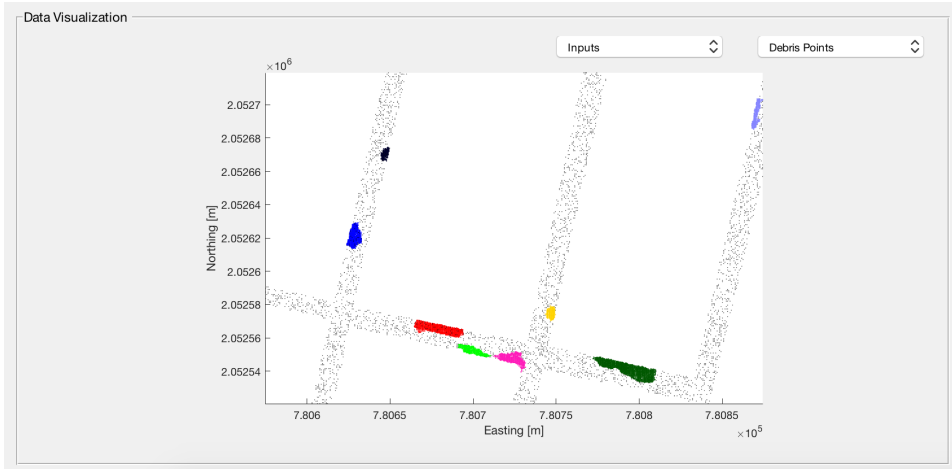


Figure 18: Visualizing the detected debris piles as points

The debris points selection will display the individual debris piles reconstructed as alpha shapes on top of the roadway points (displayed in black). The user can zoom, pan, and rotate in 3D.

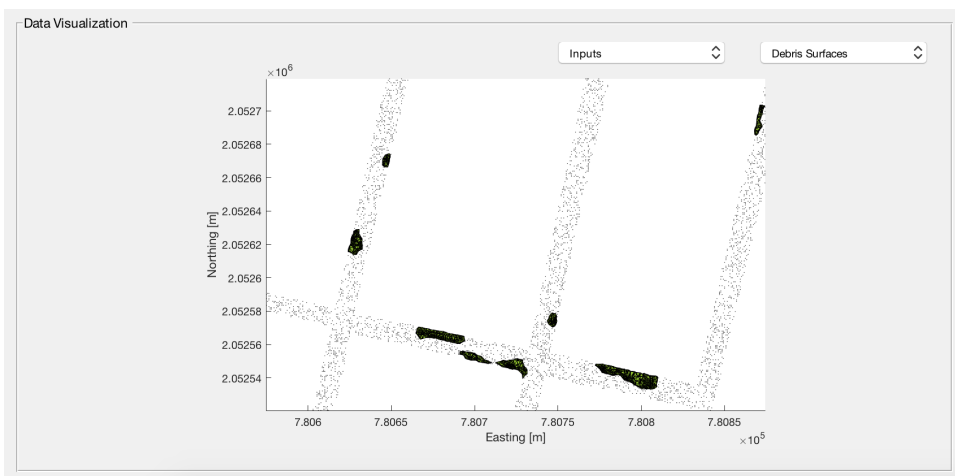


Figure 19: Visualizing the detected debris piles as alpha shapes

5.4 The Run Settings Panel

The run settings panel (the box labeled 4 in Figure 8) is where the user can change the algorithm parameters prior to running (optional), and then actually run the algorithms.

The run settings panel is divided into four subsections: (1) the units panel, (2) the algorithm parameters, (3) the restore defaults button, and (4) the run button.

5.4.1 The Units Panel

This panel allows the user to choose what units they would like to use for the minimum height and minimum volume parameters. Whichever unit is selected when hitting the run button is what will be applied to the parameters in the edit boxes. It should be noted that the choice of units for the parameters doesn't change the units in the output shapefile.

Figure 20: The user can choose between meters and feet/yards

5.4.2 The Algorithm Parameters

This set of edit boxes is used to manually input parameters for the algorithm if desired. The min. debris height allows the user to ignore debris that is below a certain height above ground. The default is 0.15m which proved to be adequate for all tested input data during development.

The min. debris volume allows the user to reject debris piles below a certain volume. Very small debris piles are typically insignificant and can sometimes be false positives from cars.

The UTM Zone is required to create a shapefile that is properly georeferenced. The default zone is 18 because that is the zone that New York City and Haiti are both in. The user can enter any valid zone (1-60).

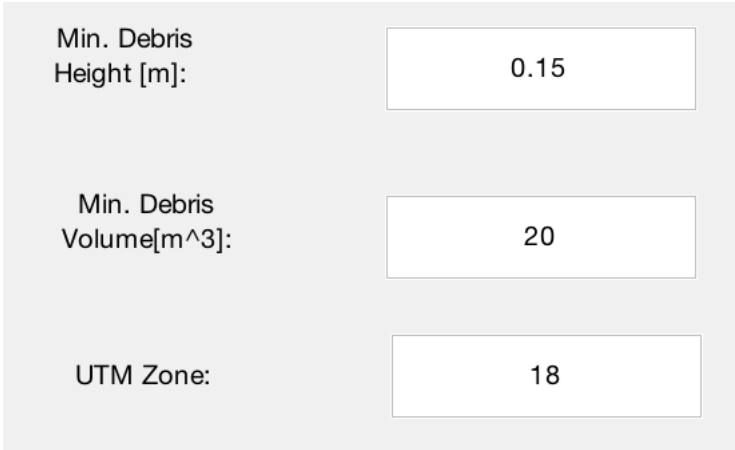


Figure 21: The algorithm parameters can be edited using these three edit boxes

5.4.3 The Restore Defaults Button

The restore defaults button restores the units to meters, the min. debris height to 0.15m, the min. debris volume to 20m^3, and the UTM zone to 18.

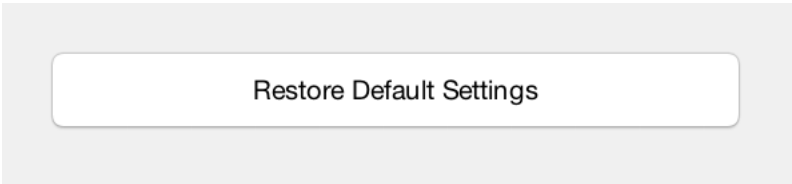


Figure 22: The Restore Defaults button

5.4.4 The Run Button

The run button will become clickable when both a valid LAS and SHP file have been loaded (not just entered) and a valid output directory has been selected. Pressing the run button launches execution of the algorithms with the parameters and units that appear on the screen. A progress bar will appear to let the user know how far along the algorithms are. As part of the algorithm execution, two shapefiles will be saved to the selected output directory. The shapefiles are named based off of the input LAS file. For example, if the input LAS file is named "haiti1.las", the output shapefiles will be "haiti1_debris_output.shp" and "haiti1_road_output.shp". The debris shapefile contains the debris polygons as well as their volumes and passable widths. The road shapefile contains the polygons of the roads

that contain debris as well as the minimum passable width and total volume of debris for the road.

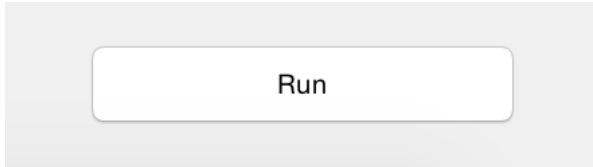


Figure 23: The Run Button

5.5 Running an Example

This section will run through an example of single file mode using real data. For this example, a folder called "files" will be used as the source of input data (see Figure 24).

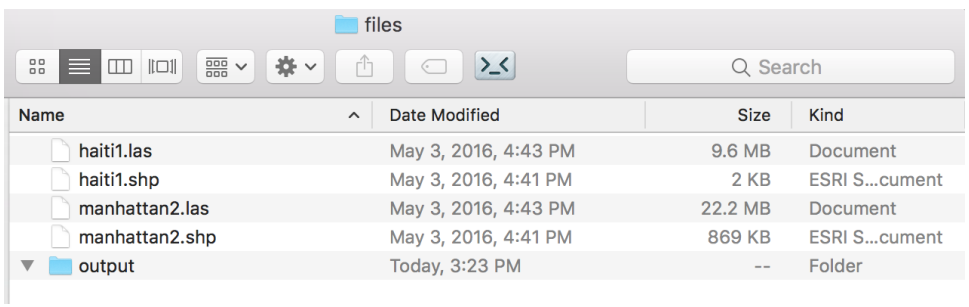


Figure 24: Input file folder for example

In this example, the files "haiti1.las" and "haiti1.shp" will be used as inputs and the outputs will be stored in "files/output".

1. Click Single File Mode in the main menu.
2. Click the browse button in the LAS section and select "haiti1.las". Hit the load button. A dialog box will pop up to let you know the file is loading.
3. Click the browse button in the SHP section and select "haiti1.shp". Hit the load button. A dialog box will pop up to let you know the file is loading.
4. Click the browse button in the Output Directory section and click the "output" folder. The run button should now become clickable.
5. Go to the inputs menu in the data visualization panel and select "Both". The screen should now look like it does in Figure 25.

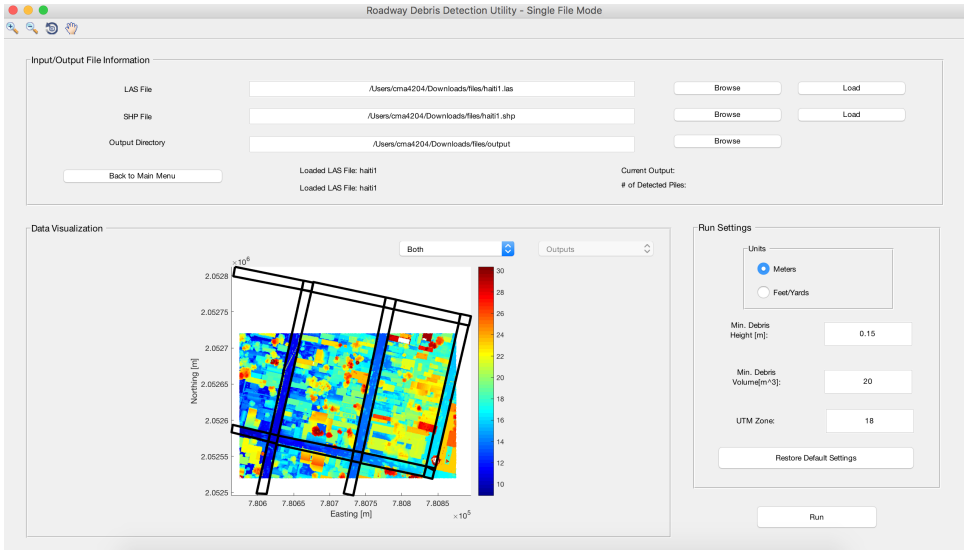


Figure 25: What the application should look like after selecting the inputs and displaying the combined point cloud and road network.

6. The default parameters are perfectly fine for these data, so click the run button. A progress bar will display while the algorithm is running and then disappear once it has finished.

7. Change the data visualization to the reconstructed debris pile surfaces by changing the output menu to "Debris Surfaces". Zoom in to look at the debris surfaces (see Figure 26).

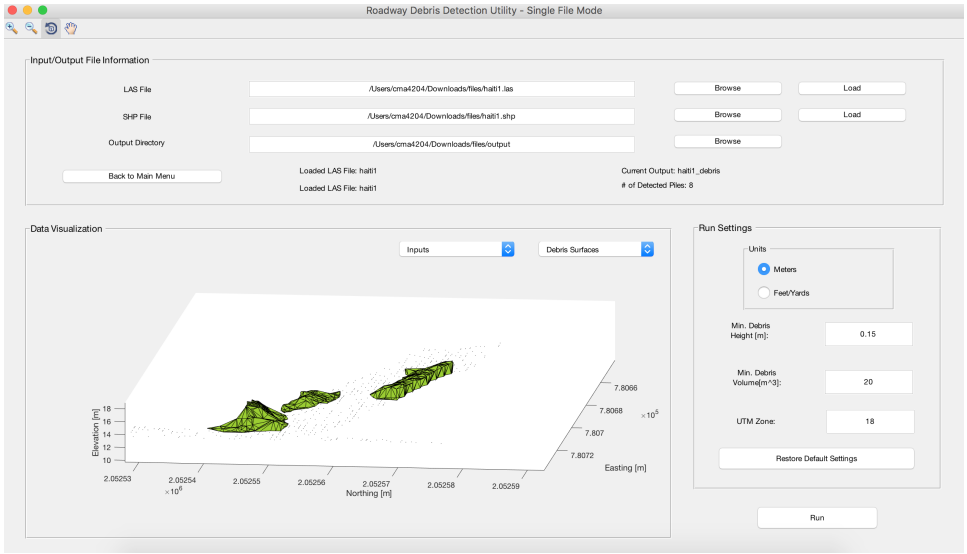
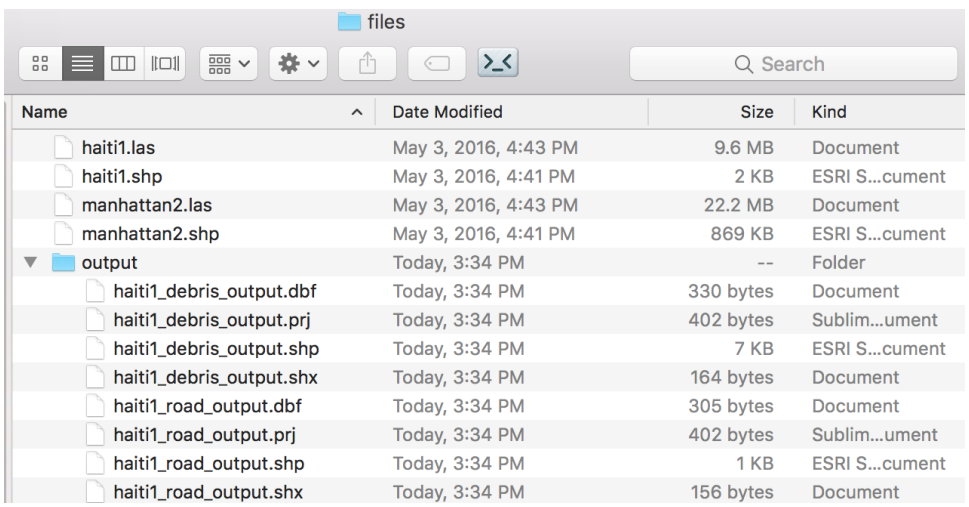


Figure 26: What the application should look like after running the algorithm and zooming in on the debris surfaces

8. Next, look in the output directory. The debris and road shapefiles should be there.



files

Search

Name	Date Modified	Size	Kind
haiti1.las	May 3, 2016, 4:43 PM	9.6 MB	Document
haiti1.shp	May 3, 2016, 4:41 PM	2 KB	ESRI S...cument
manhattan2.las	May 3, 2016, 4:43 PM	22.2 MB	Document
manhattan2.shp	May 3, 2016, 4:41 PM	869 KB	ESRI S...cument
output	Today, 3:34 PM	--	Folder
haiti1_debris_output.dbf	Today, 3:34 PM	330 bytes	Document
haiti1_debris_output.prj	Today, 3:34 PM	402 bytes	Sublim...ument
haiti1_debris_output.shp	Today, 3:34 PM	7 KB	ESRI S...cument
haiti1_debris_output.shx	Today, 3:34 PM	164 bytes	Document
haiti1_road_output.dbf	Today, 3:34 PM	305 bytes	Document
haiti1_road_output.prj	Today, 3:34 PM	402 bytes	Sublim...ument
haiti1_road_output.shp	Today, 3:34 PM	1 KB	ESRI S...cument
haiti1_road_output.shx	Today, 3:34 PM	156 bytes	Document

Figure 27: The debris and road shapefiles appear in the output directory after a successful run

9. The output files are in the output directory as expected and can be viewed in any GIS program. Congratulations, the example is complete.

6 The Batch File Mode Interface

The batch file mode interface pops up when selecting batch file mode from the main menu. This interface contains all of the controls and features needed to perform debris detection, volume estimation, and shapefile outputs for all point cloud/road network file pairs in a single directory. The interface is divided into two sections: (1) the input/output directory panel and (2) the run settings panel. The interface with the two sections labeled is displayed in Figure 28. Each of these two sections and their components will be described separately.

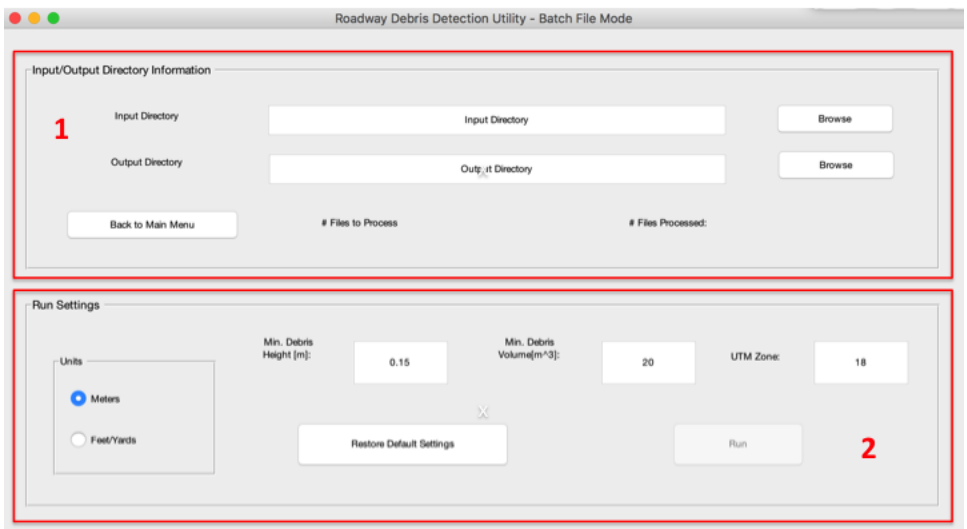


Figure 28: The Batch File Mode Interface

6.1 The Input/Output File Panel

The input/output file panel (the box labeled 1 in Figure 28) is where the user selects the input and output directories for batch file mode. Additionally, the user can choose to return to the main menu from this panel. Once an input and output directory have been selected, the "Run" button will become clickable and the algorithms can be executed. For the sake of this manual, the input/output panel can be divided into four subsections: (1) the input directory section, (2) the output directory section, (3) the back button, and (4) the text display.

6.1.1 The Input Directory Section

The Input Directory section contains all of the tools needed to select a directory in which the input point cloud LAS files (*.las) and road network shapefiles (*.shp) exist. The text edit bar allows the user to manually enter a directory by typing or copying and pasting the path. If the entered path doesn't lead to a valid directory, an error will appear. The browse button lets the user navigate to an input directory At any point, the user can select a new input directory. The point cloud files and the road network shapefiles MUST have the same name. The application searches the folder for LAS and SHP files that are named the same and will run on all of the matching pairs. For example, "haiti1.las" and "haiti1.shp" would be detected as a valid pair whereas "debris1.las" and"roads1.shp" would not. It should be noted that the LAS and SHP file must both be in the same projection (UTM).

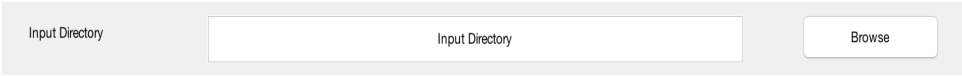


Figure 29: The Input Directory section

6.1.2 The Output Directory Section

The Output Directory section contains all of the tools needed to select a directory in which the output shapefiles will be saved. The text edit bar allows the user to manually enter a directory by typing or copying and pasting the path. If the entered path doesn't lead to a valid directory, an error will appear. The browse button lets the user navigate to an output directory At any point, the user can select a new output directory.

Figure 30: The Output Directory section

6.1.3 The Back Button

The back button serves one simple purpose - it brings the user back to the main menu. The back button should only be used when the user wants to switch running modes (from single to batch or vice versa).

Figure 31: The Back Button

6.1.4 The Text Display

The text display lets the user know how many LAS-SHP file pairs were detected by displaying the number in the "# Files to Processs:" section. The display gets updated as soon as the user hits the run button. The "# Files Processed:" section starts at zero and increments by one after each file pair is successfully processed.

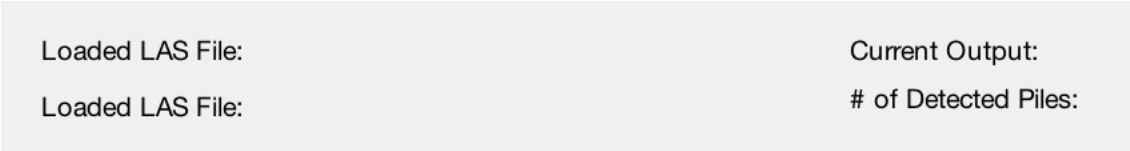


Figure 32: The Text Display

6.2 The Run Settings Panel

The run settings panel (the box labeled 2 in Figure 28) is where the user can change the algorithm parameters prior to running (optional), and then actually run the algorithms. The run settings panel is divided into four subsections: (1) the units panel, (2) the algorithm parameters, (3) the restore defaults button, and (4) the run button.

6.2.1 The Units Panel

This panel allows the user to choose what units they would like to use for the minimum height and minimum volume parameters. Whichever unit is selected when hitting the run button is what will be applied to the parameters in the edit boxes. It should be noted that the choice of units for the parameters doesn't change the units in the output shapefile.

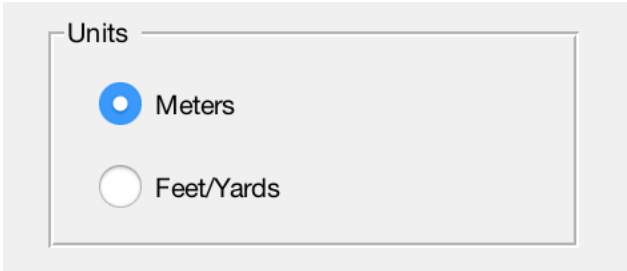


Figure 33: The user can choose between meters and feet/yards

6.2.2 The Algorithm Parameters

This set of edit boxes is used to manually input parameters for the algorithm if desired. The min. debris height allows the user to ignore debris that is below a certain height above ground. The default is 0.15m which proved to be adequate for all tested input data during development.

The min. debris volume allows the user to reject debris piles below a certain volume. Very small debris piles are typically insignificant and can sometimes be false positives from cars.

The UTM Zone is required to create a shapefile that is properly georeferenced. The default zone is 18 because that is the zone that New York City and Haiti are both in. The user can enter any valid zone (1-60).

The parameters in the application when hitting the run button will be used for all valid files in the input directory.

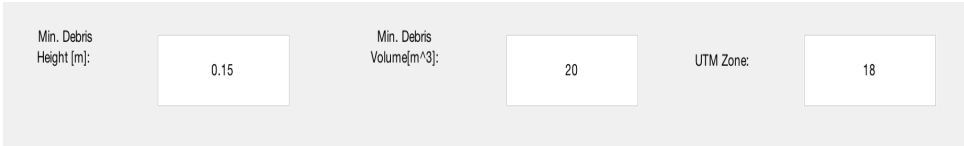


Figure 34: The algorithm parameters can be edited using these three edit boxes

6.2.3 The Restore Defaults Button

The restore defaults button restores the units to meters, the min. debris height to 0.15m, the min. debris volume to 20m^3, and the UTM zone to 18.

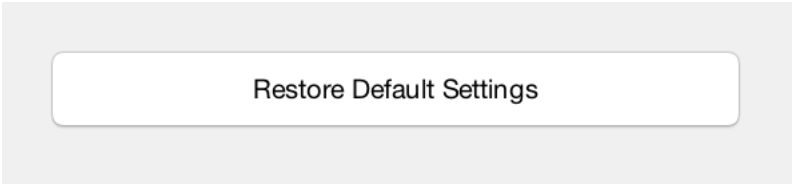


Figure 35: The Restore Defaults button

6.2.4 The Run Button

The run button will become clickable when both a valid input and output directory have been selected. Pressing the run button launches execution of the algorithms with the parameters and units that appear on the screen. A progress bar will appear to let the user know approximately how long it will take to finish processing all of the files in directory. As part of the algorithm execution, two shapefiles will be saved to the selected output directory for each input file processed. The shapefiles are named based off of the input LAS file. For example, if the input LAS file is named "haiti1.las", the output shapefiles will be "haiti1_debris_output.shp" and "haiti1_road_output.shp". The point cloud files and the road network shapefiles MUST have the same name. The application searches the folder for LAS and SHP files that are named the same and will run on all of the matching pairs. For example, "haiti1.las" and "haiti1.shp" would be detected as a valid pair whereas "debris1.las" and"roads1.shp" would not. The debris shapefile contains the debris polygons as well as their volumes and passable widths. The road shapefile contains the polygons of the roads that contain debris as well as the minimum passable width and total volume of debris for the road.

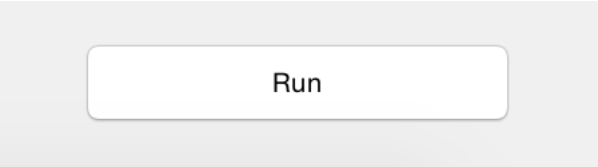


Figure 36: The Run Button

6.3 Running an Example

This section will run through an example of batch file mode using real data. For this example, a folder called "files" will be used as the source of input data (see Figure 37).

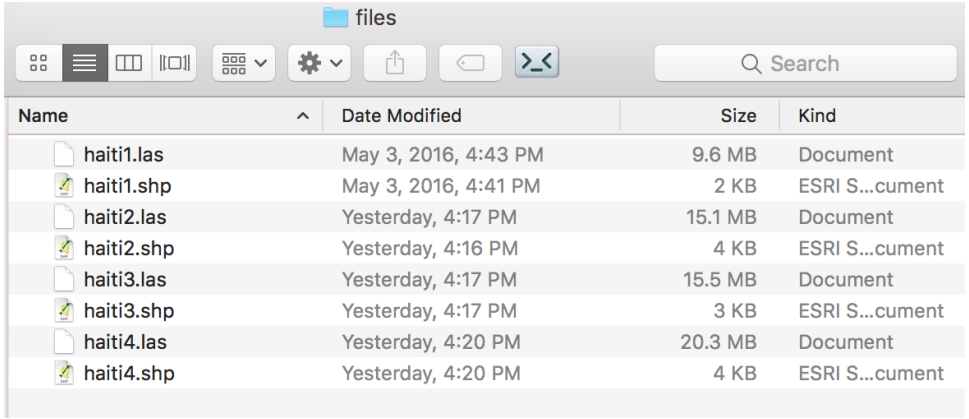


Figure 37: Input file folder for batch file mode example

In this example, the files "haiti1.las" through "haiti4.las" and the corresponding shapefiles, "haiti1.shp" through "haiti4.shp" will be used as inputs and the outputs will be stored in "files/output".

1. Click Batch File Mode in the main menu.
2. Click the browse button in the Input Directory section and navigate to the "files" folder.
3. Click the browse button in the Output Directory section and navigate to the "files/output". The run button should now become clickable.
4. The default parameters are perfectly fine for these data, so click the run button. A progress bar will appear. The screen should now look like it does in Figure 38.

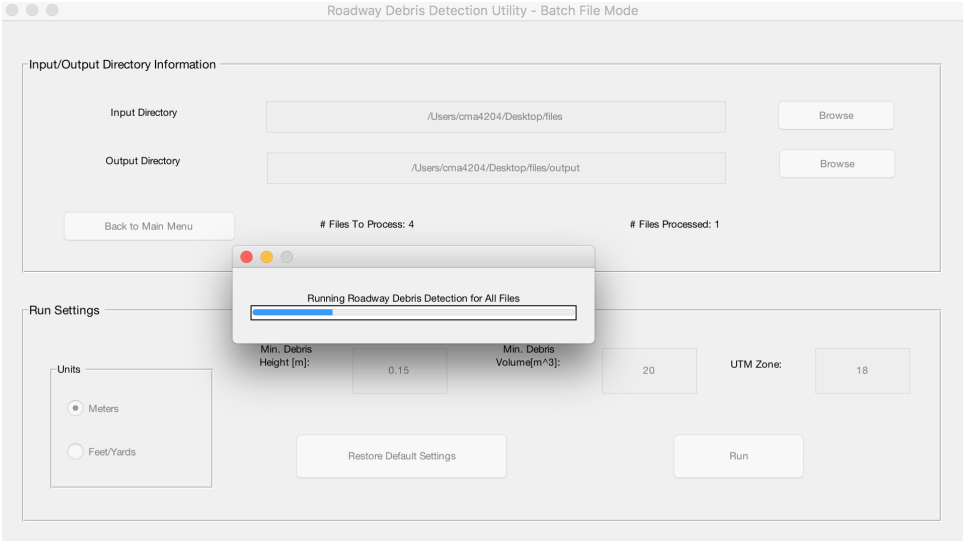
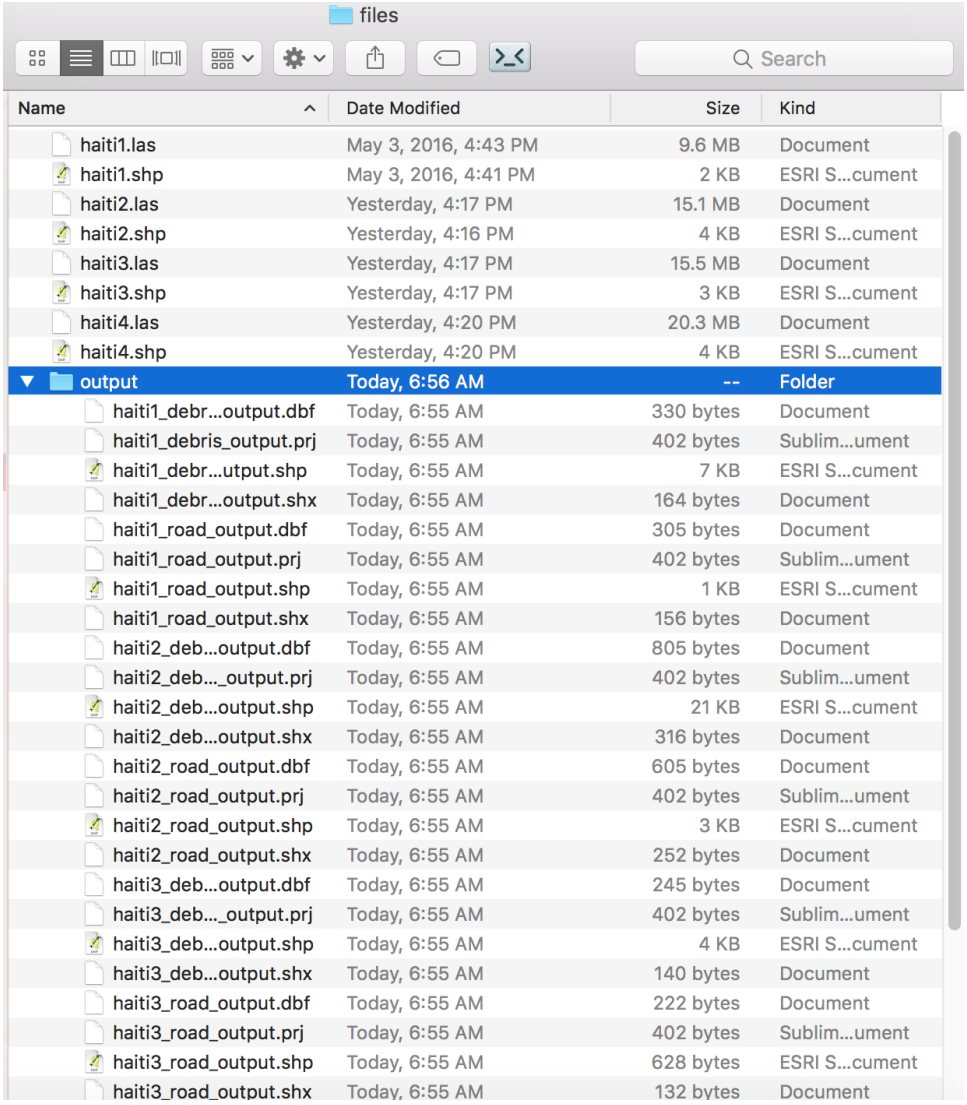


Figure 38: What the application should look while running in batch mode

5. Next, look in the output directory. The debris and road shapefiles should be there.



files

Name	Date Modified	Size	Kind
haiti1.las	May 3, 2016, 4:43 PM	9.6 MB	Document
haiti1.shp	May 3, 2016, 4:41 PM	2 KB	ESRI S...cument
haiti2.las	Yesterday, 4:17 PM	15.1 MB	Document
haiti2.shp	Yesterday, 4:16 PM	4 KB	ESRI S...cument
haiti3.las	Yesterday, 4:17 PM	15.5 MB	Document
haiti3.shp	Yesterday, 4:17 PM	3 KB	ESRI S...cument
haiti4.las	Yesterday, 4:20 PM	20.3 MB	Document
haiti4.shp	Yesterday, 4:20 PM	4 KB	ESRI S...cument
output	Today, 6:56 AM	--	Folder
haiti1_debr...output.dbf	Today, 6:55 AM	330 bytes	Document
haiti1_debris_output.prj	Today, 6:55 AM	402 bytes	Sublim...ument
haiti1_debr...utput.shp	Today, 6:55 AM	7 KB	ESRI S...cument
haiti1_debr...output.shx	Today, 6:55 AM	164 bytes	Document
haiti1_road_output.dbf	Today, 6:55 AM	305 bytes	Document
haiti1_road_output.prj	Today, 6:55 AM	402 bytes	Sublim...ument
haiti1_road_output.shp	Today, 6:55 AM	1 KB	ESRI S...cument
haiti1_road_output.shx	Today, 6:55 AM	156 bytes	Document
haiti2_deb...output.dbf	Today, 6:55 AM	805 bytes	Document
haiti2_deb..._output.prj	Today, 6:55 AM	402 bytes	Sublim...ument
haiti2_deb...output.shp	Today, 6:55 AM	21 KB	ESRI S...cument
haiti2_deb...output.shx	Today, 6:55 AM	316 bytes	Document
haiti2_road_output.dbf	Today, 6:55 AM	605 bytes	Document
haiti2_road_output.prj	Today, 6:55 AM	402 bytes	Sublim...ument
haiti2_road_output.shp	Today, 6:55 AM	3 KB	ESRI S...cument
haiti2_road_output.shx	Today, 6:55 AM	252 bytes	Document
haiti3_deb...output.dbf	Today, 6:55 AM	245 bytes	Document
haiti3_deb..._output.prj	Today, 6:55 AM	402 bytes	Sublim...ument
haiti3_deb...output.shp	Today, 6:55 AM	4 KB	ESRI S...cument
haiti3_deb...output.shx	Today, 6:55 AM	140 bytes	Document
haiti3_road_output.dbf	Today, 6:55 AM	222 bytes	Document
haiti3_road_output.prj	Today, 6:55 AM	402 bytes	Sublim...ument
haiti3_road_output.shp	Today, 6:55 AM	628 bytes	ESRI S...cument
haiti3_road_output.shx	Today, 6:55 AM	132 bytes	Document

Figure 39: The debris and road shapefiles appear in the output directory after a successful run

6. The output files are in the output directory as expected and can be viewed in any GIS program. Congratulations, the example is complete.

APPENDIX H:
Flood Utility User Manual

FloodUtility v1.0 User Manual

September 27, 2016

Contents

1 Overview **2**

2 Installation **3**

3 Running the Application **6**

4 The Main Menu Interface **7**
- 4.1 Single File Mode . . . 7
- 4.2 Batch File Mode . . . 8

5 The Single File Mode Interface **8**
- 5.1 The Toolbar . . . 9
  - 5.1.1 Zoom In . . . 9
  - 5.1.2 Zoom Out . . . 9
  - 5.1.3 Pan . . . 9
  - 5.1.4 Pick a Point . . . 10
  - 5.1.5 Delete a Point . . . 10
- 5.2 The Input/Output File Panel . . . 10
  - 5.2.1 The Image File Section . . . 10
  - 5.2.2 The SHP Section . . . 11
  - 5.2.3 The DEM Section . . . 11
  - 5.2.4 The Output Directory Section . . . 12
  - 5.2.5 The Back Button . . . 12
  - 5.2.6 The Text Display . . . 13
- 5.3 The Data Visualization Panel . . . 13
  - 5.3.1 The Inputs Dropdown Menu . . . 13
  - 5.3.2 The Outputs Dropdown Menu . . . 16
- 5.4 The Run Settings Panel . . . 20

5.4.1 The Algorithm Parameters . . . 21
5.4.2 The Run Method Panel . . . 22
5.4.3 The Restore Defaults Button . . . 23
5.4.4 The Run Button . . . 23
5.5 Running an Example . . . 24

6 The Batch File Mode Interface 26
6.1 The Toolbar . . . 27
6.1.1 Zoom In . . . 28
6.1.2 Zoom Out . . . 28
6.1.3 Pan . . . 28
6.1.4 Pick a Point . . . 28
6.1.5 Delete a Point . . . 29
6.2 The Input/Output File Panel . . . 29
6.2.1 The Input Directory Section . . . 29
6.2.2 The Output Directory Section . . . 30
6.2.3 The Back Button . . . 30
6.2.4 The Text Display . . . 30
6.3 The Data Visualization Panel . . . 31
6.3.1 The Inputs Dropdown Menu . . . 31
6.3.2 The Outputs Dropdown Menu . . . 32
6.4 The Run Settings Panel . . . 36
6.4.1 The Algorithm Parameters . . . 37
6.4.2 The Run Method Panel . . . 38
6.4.3 The Restore Defaults Button . . . 38
6.4.4 The Run Buttons . . . 39
6.5 Running an Example . . . 39

1 Overview

This is the user manual for the FloodUtility v1.0 application. The application provides a user interface for generating roadway flood maps using commercial remote sensing (CRS) data. The underlying algorithms take an input flood image (*.tif/tiff), a corresponding road network shapefile (*.shp), and a digital elevation model (DEM) raster (*.tif/tiff) and finds water in the roadways and estimates its depth. A shapefile is generated containing roadway flood polygons that contain the water depth. The manual covers everything from installation to outputs and describes the function of every feature and component of the application. Screenshots are included with the descriptions as well as a start-to-finish example so that the user can follow along and get a feel for flow of the application.

2 Installation

This section takes the user step-by-step through the installation process.

1. Begin by running the MyAppInstaller_web application. This will open the FloodUtility Installer, a step-by-step application that will guide you through the installation process. Note that an internet connection is required to install the application. If your internet connection requires a proxy server (this is rarely the case), configure that by hitting the "Connection Settings" button. Otherwise, hit the next button.

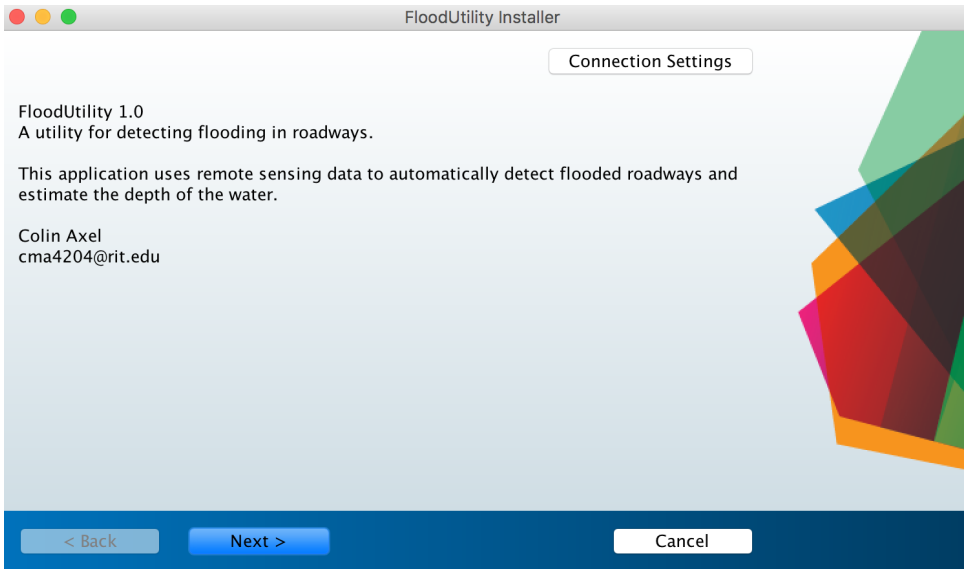


Figure 1: The FloodUtility Installer window

2. The next window that will pop up is the Installation Options window. Choose an installation directory using the browse button, and then hit next.

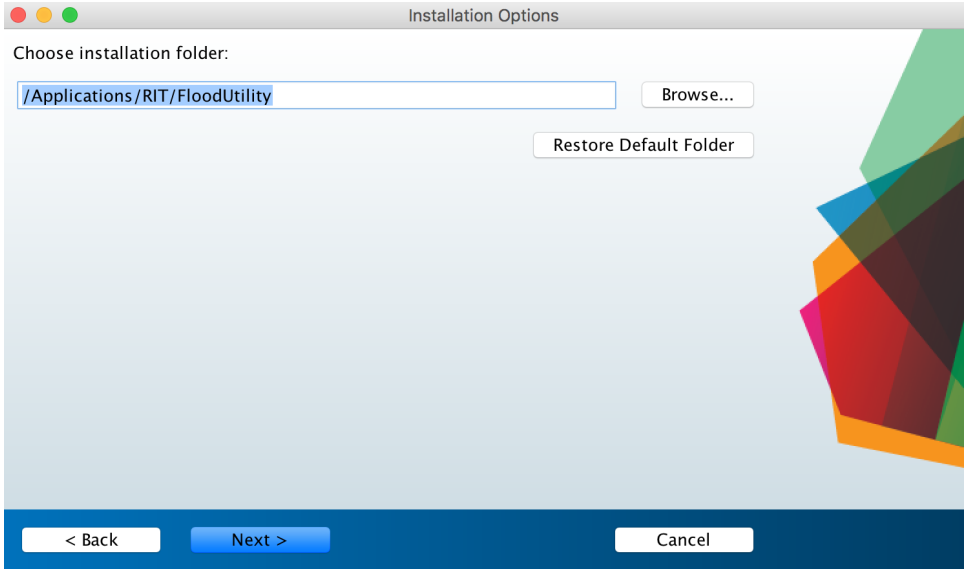


Figure 2: The Installation Options window

3. The Required Software window will inform you that the MATLAB Runtime software is required. The application cannot run without this piece of software, so install it either in the default directory or another directory of your choice. If MATLAB Runtime is already installed on your computer it will automatically be detected. Hit the next button.

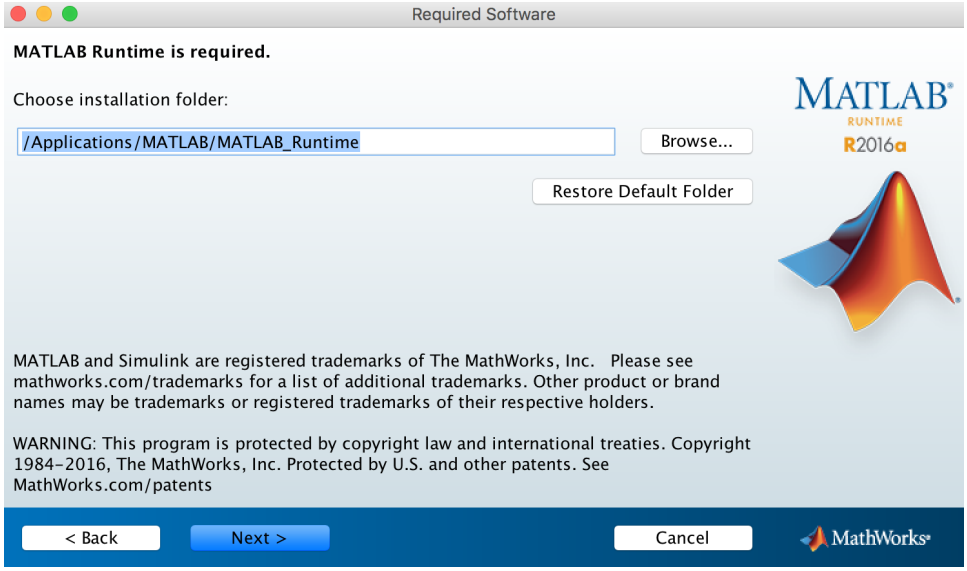


Figure 3: The Installation Options window

4. The Confirmation window will pop up and provide some basic information about the MATLAB Runtime and where it will be installed. Hit the Install > button.

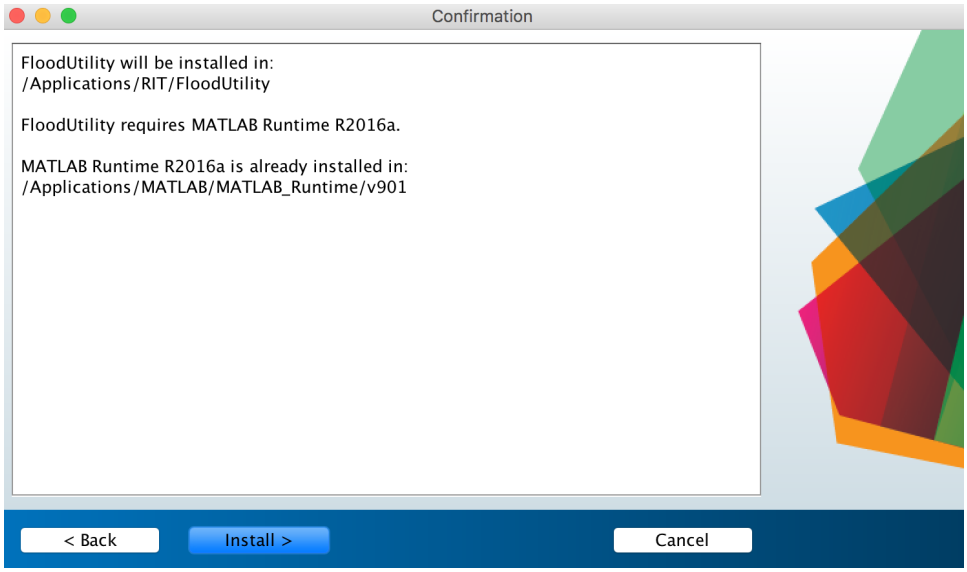


Figure 4: The Confirmation window

5. The installer will take a few moments to install and then the Installation Complete window will appear, letting you know that the installation completed successfully. Hit the finish button. At this point the application is installed and ready to use.

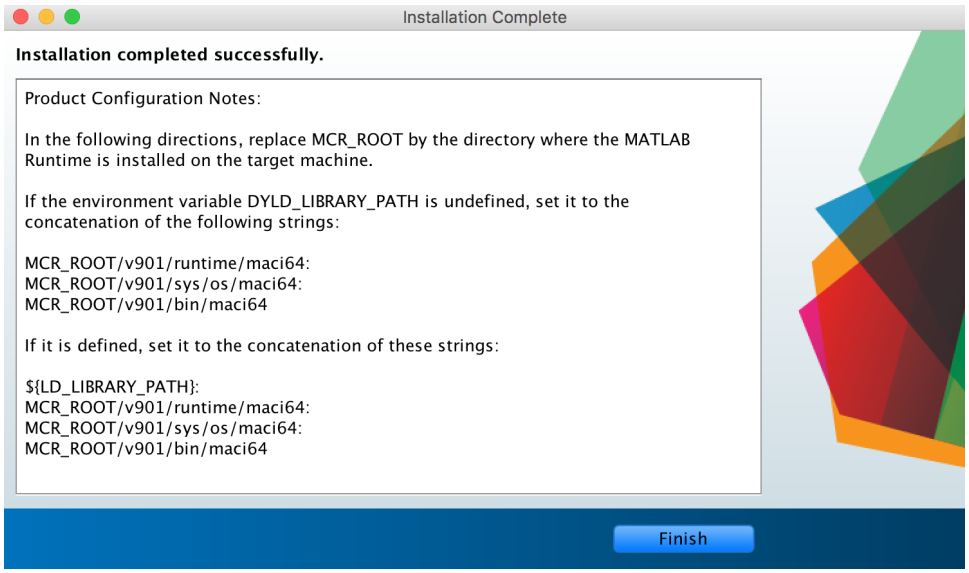


Figure 5: The Installation Complete window

3 Running the Application

The location of the application will depend on the directory that was selected during installation. For example, the path on this machine is: "/Applications/RIT/FloodUtility". Navigate to that directory and location the "application" folder. Inside of this folder you will see the FloodUtility application. Double click the application to run it.

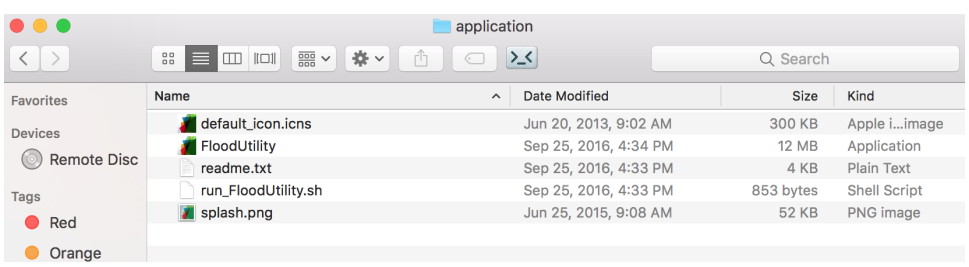


Figure 6: Locating the FloodUtility application

4 The Main Menu Interface

The first window that pops up when running the application is the main menu. The main menu is a simple interface that presents the user with two run mode options: Single File Mode and Batch File Mode.

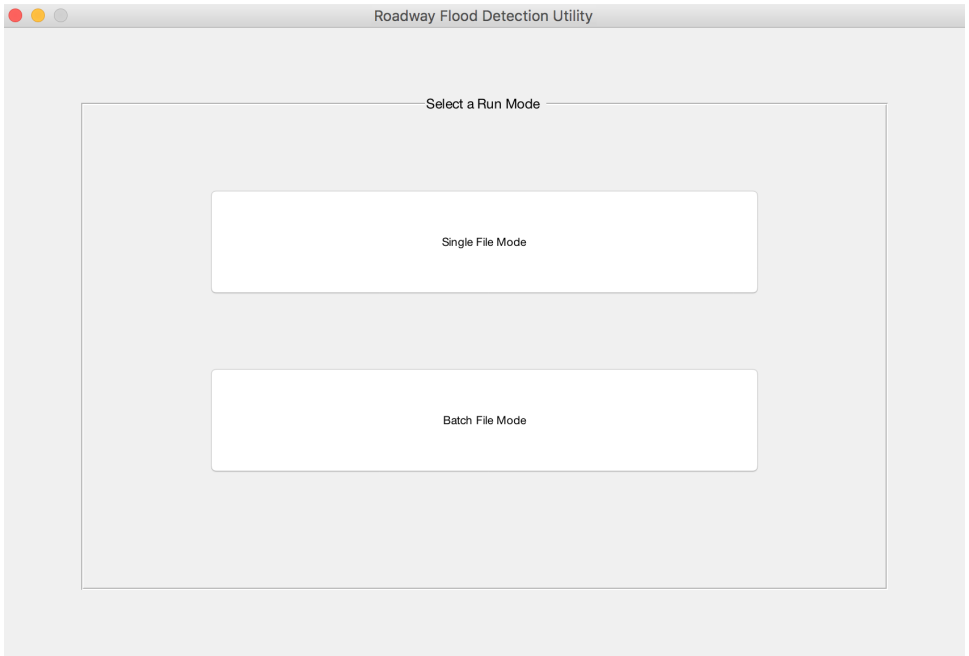


Figure 7: The main menu interface

4.1 Single File Mode

Single file mode is used to perform roadway flood detection and depth estimation on a single flood image and requires a single flood image (*.tif/*.tiff), road network shapefile (*.shp), and DEM image (*.tif/*.tiff). Single file mode requires the user to manually load one file at a time, adjust settings (optional), and then run the algorithm. This mode is best to use when you want to be able to manually adjust settings and view the results for a single file at a time.

4.2 Batch File Mode

Batch file mode is used to perform roadway flood detection and depth estimation on all images in a selected directory and requires corresponding flood images (*.tif/*.tiff), road network shapefiles (*.shp), and DEM images (*.tif/*.tiff), to be all numbered the same. For example: "flood1.tif", "roads1.tif", and "dem1.tif" or "nyc_f2.tif", "nyc_r2.tif", and "nyc_d2.tif". Batch file mode requires the user to select an input directory (that contains the files to run), adjust settings (optional), and then run the algorithm for all files in the input directory. This mode is best to use when you want to let the program run for several files without needing any user interaction other than the initial setup.

5 The Single File Mode Interface

The single file mode interface pops up when selecting single file mode from the main menu. This interface contains all of the controls and features needed to perform flood detection, depth estimation, and shapefile output on a single flood scene. The interface is divided into four sections: (1) the toolbar, (2) the input/output file panel, (3) the data visualization panel, and (4) the run settings panel. The interface with the four sections labeled is displayed in Figure 8. Each of these four sections and their components will be described in separate subsections.

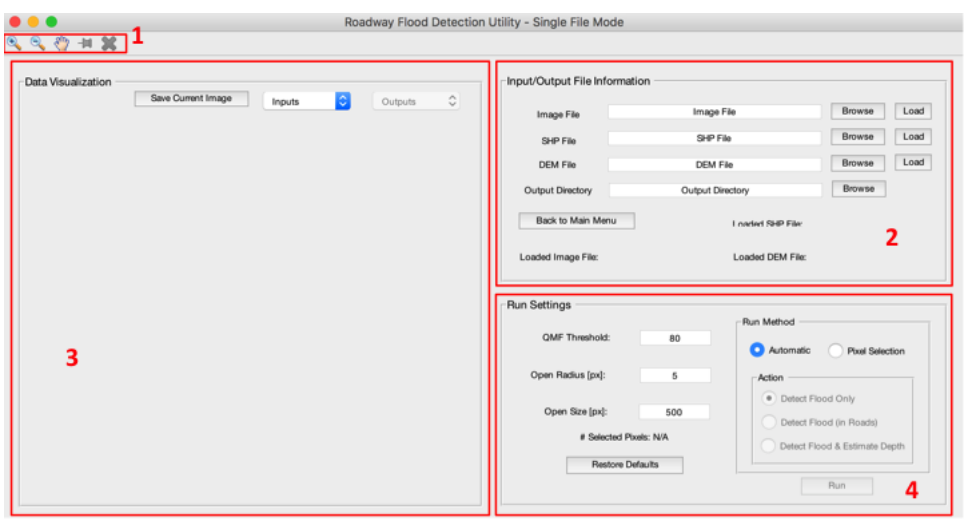


Figure 8: The single file mode interface

5.1 The Toolbar

The toolbar (the box labeled 1 in Figure 8) contains five tools thats can be used when viewing flood images, road networks, or DEM images in the data visualization panel. Different input and outputs can be visualized by selecting one of the options from the input or output menu. Once a feature has appeared in the visualization panel, the icons on the toolbar let you do three things: zoom in, zoom out, and pan. If the feature is the flood image, two more tools are also available: pick a point and delete a point. The five tools will all be described in their own subsections.

(a) (b) (c) (d) (e)

Figure 9: The five tools of the toolbar: (a) zoom in (b) zoom out (c) pan (d) pick a point (e) delete a point

5.1.1 Zoom In

This tool allows the user to zoom in when viewing a flood image, road network, or DEM. Sometimes images are very large and it is hard to see what is going on when viewing the whole image. In these cases, use the zoom in tool to view a specific region. When the tool is selected, click the image to zoom into a specific point or click and drag a box to zoom into a whole region.

5.1.2 Zoom Out

This tool allows the user to zoom out when viewing a flood image, road network, or DEM. This tool only works after the screen has already been zoomed in. When the tool is selected, click the image anywhere to zoom out.

5.1.3 Pan

This tool allows the user to navigate the display after it has been zoomed in. When the tool is selected, click the image and drag.

5.1.4 Pick a Point

This tool allows a user to manually select flood pixels in a flood image. The tool becomes clickable when the data visualization panel is showing the flood image and the "pixel selection" button has been clicked in the Run Settings panel. To select a pixel, simply click the pick a point tool and then click a flood pixel in the image. Multiple pixels can be selected by repeating this process (the tool must be clicked again to select a new point). Every time a pixel is selected, the "Selected Pixels" counter in the Run Settings panel will increase by one.

5.1.5 Delete a Point

This tool allows a user to delete selected flood pixels in a flood image. The tool becomes clickable as soon as a single flood pixels has been selected. To delete a point, simply click the tool. Clicking the tool will only delete the most recently selected point. The tool can be clicked as long as there are still points present (the points will be visible on the image screen). Every time a point is deleted, the "Selected Pixels" counter in the Run Settings panel will decrease by one.

5.2 The Input/Output File Panel

The input/output file panel (the box labeled 2 in Figure 8) is where the user selects and loads the input flood image, road network shapefile, DEM image, and output directory (for the output shapefiles). Additionally, the user can choose to return to the main menu from this panel. As files are loaded, the three run modes in the Run Settings panel will become clickable. The input images and shapefiles can be in either latitude/longitude or UTM coordinates. For the sake of this manual, the input/output panel can be divided into five subsections: (1) the image file section, (2) the SHP section, (3) the DEM section, (4) the Output Directory section (4), the back button, and (5) the text display.

5.2.1 The Image File Section

The image file section contains all of the tools needed to select and load a flood image GeoTIFF. The text edit bar allows the user to manually enter the filename (including path and extension) of a GeoTIFF (*.tif/*.tiff) file either by typing or copying and pasting the path. If the entered path doesn't lead to a valid GeoTIFF file, an error will appear. Additionally, if the flood image doesn't have at least three spectral bands, an error will appear. This algorithm is not designed for grayscale imagery. The browse button lets the user navigate to a GeoTIFF file, and will also prevent the user from selecting a file that

isn't a valid GeoTIFF file. Once a valid file has been selected (if the filename appears in the text edit bar it means it is valid) the user can then load the file by simply hitting the load button. Once a file has been loaded it can be viewed in the data visualization panel. At any point, the user can select and load a new flood image file (note that it won't be loaded until the load button is clicked). Once a flood image file is loaded, the filename will appear in front of the "Loaded Image File:" section of the text display.

Figure 10: The Image File section

5.2.2 The SHP Section

The SHP section contains all of the tools needed to select and load a road network shapefile. The text edit bar allows the user to manually enter the filename (including path and extension) of a SHP (*.shp) file either by typing or copying and pasting the path. If the entered path doesn't lead to a valid shapefile, an error will appear. The browse button lets the user navigate to a shapefile, and will also prevent the user from selecting a file that isn't a valid shapefile. Once a valid file has been selected (if the filename appears in the text edit bar it means it is valid) the user can then load the file by simply hitting the load button. Once a file has been loaded it can be viewed in the data visualization panel. At any point, the user can select and load a new shapefile (note that it won't be loaded until the load button is clicked). Once a shapefile is loaded, the filename will appear in front of the"Loaded SHP File:" section of the text display.

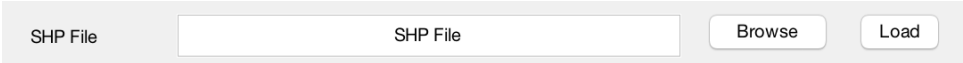


Figure 11: The SHP section

5.2.3 The DEM Section

The image file section contains all of the tools needed to select and load a DEM image GeoTIFF. The text edit bar allows the user to manually enter the filename (including path and extension) of a GeoTIFF (*.tif/*.tiff) file either by typing or copying and pasting the path. If the entered path doesn't lead to a valid GeoTIFF file, an error will appear. Additionally, if the DEM image has more than one band, an error will appear. DEM rasters should be grayscale images in which each pixel represents an elevation. The browse button lets the user navigate to a GeoTIFF file, and will also prevent the user from selecting a file

that isn't a valid GeoTIFF file. Once a valid file has been selected (if the filename appears in the text edit bar it means it is valid) the user can then load the file by simply hitting the load button. Once a file has been loaded it can be viewed in the data visualization panel. At any point, the user can select and load a new DEM image file (note that it won't be loaded until the load button is clicked). Once a DEM image file is loaded, the filename will appear in front of the "Loaded DEM File:" section of the text display.

Figure 12: The DEM File section

5.2.4 The Output Directory Section

The Output Directory section contains all of the tools needed to select a directory in which the output shapefiles will be saved. The text edit bar allows the user to manually enter a directory by typing or copying and pasting the path. If the entered path doesn't lead to a valid directory, an error will appear. The browse button lets the user navigate to an output directory At any point, the user can select a new output directory.

Figure 13: The Output Directory section

5.2.5 The Back Button

The back button serves one simple purpose - it brings the user back to the main menu. The back button should only be used when the user wants to switch running modes (from single to batch or vice versa). All currently loaded data will be cleared when hitting the back button (except for the saved shapefile, it will remain in the selected output directory).

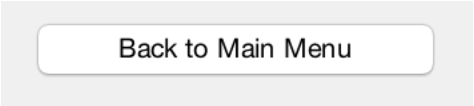


Figure 14: The Back Button

5.2.6 The Text Display

The text display exists mainly so that the user can reference what files are currently loaded into the application. The "Loaded Image File:" display lets the user know which flood image file is currently loaded - this file is what will be used for visualization and running (note: even if a new file has been selected, until it has been loaded the old file will remain). The "Loaded SHP File:" display lets the user know which SHP file is currently loaded - this file is what will be used for visualization and running (note: even if a new file has been selected, until it has been loaded the old file will remain). The "Loaded DEM File:" display lets the user know which DEM image file is currently loaded - this file is what will be used for visualization and running (note: even if a new file has been selected, until it has been loaded the old file will remain).

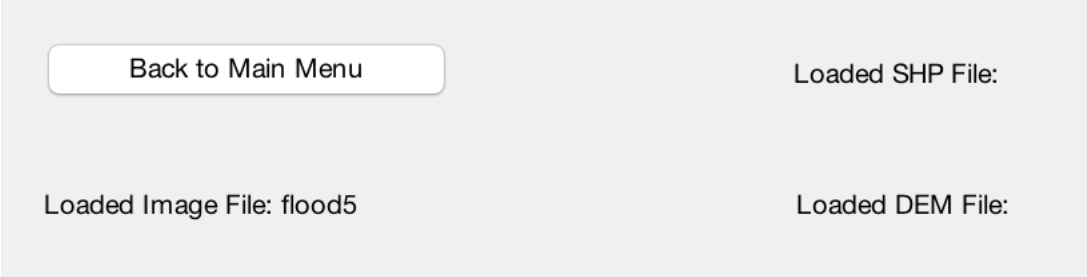


Figure 15: The Text Display section

5.3 The Data Visualization Panel

The data visualization panel (the box labeled 3 in Figure 8) is where the user can visualize the inputs and outputs (once generated). The data visualization panel doesn't become activated until the user chooses a input or output to view from the dropdown menus. The rest of this section will give examples of what the data visualization panel looks like for the various inputs/outputs.

5.3.1 The Inputs Dropdown Menu

The inputs dropdown menu lets the user choose from three different choices: image, roads, and DEM. If the user tries to visualize a file that hasn't been loaded yet, an error message will pop up.

The image selection will display the flood image. The user can zoom, pan, and select/delete a point (if in pixel selection mode).

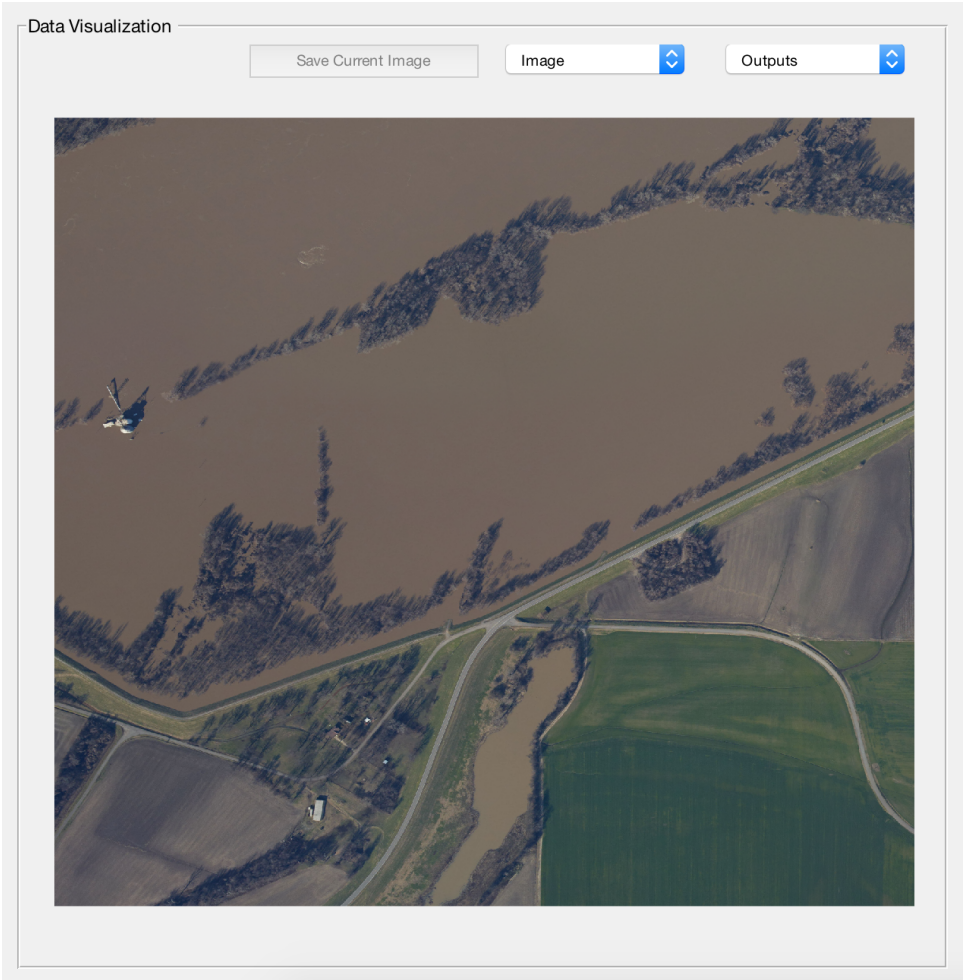


Figure 16: Visualizing the flood image

The roads selection will display the road network. The user can zoom and pan.

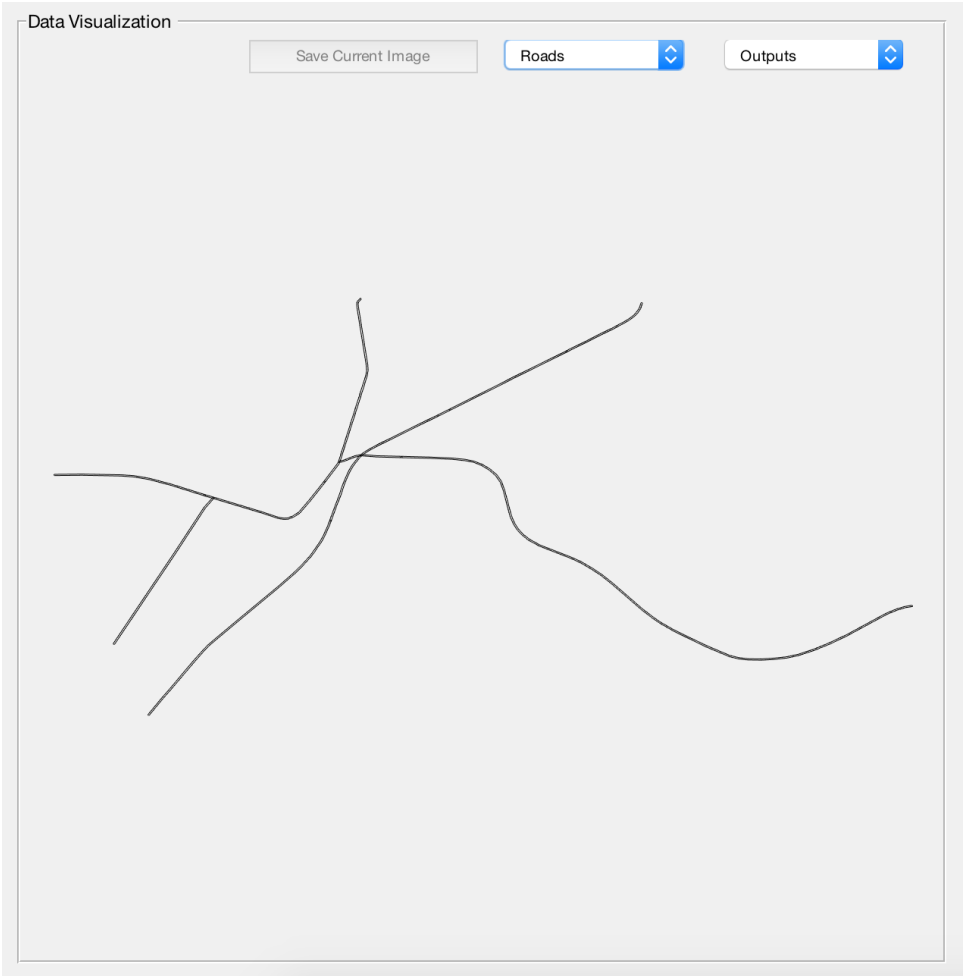


Figure 17: Visualizing the road network

The DEM selection will display the digital elevation model image. The colorbar on the right indicates the elevation scale (in whatever units the input DEM is). The user can zoom and pan.

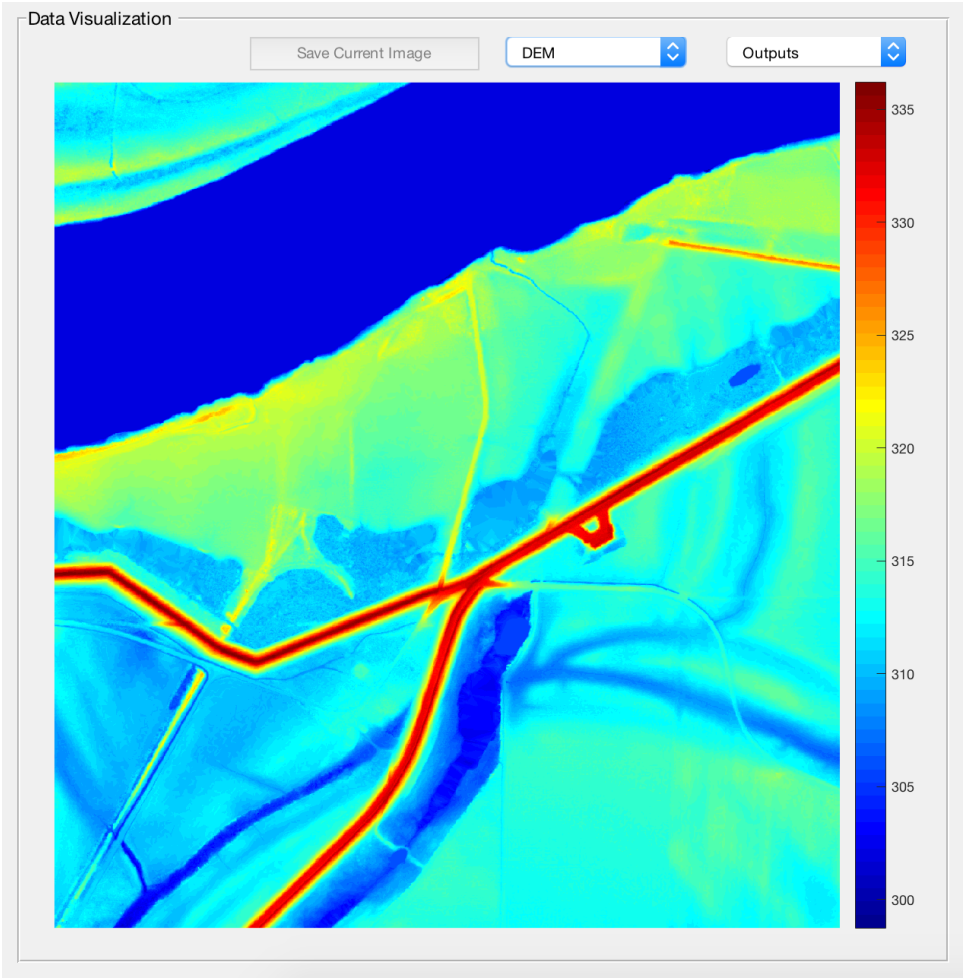


Figure 18: Visualizing the digital elevation model

5.3.2 The Outputs Dropdown Menu

The outputs dropdown menu lets the user choose from four different choices: flood map, road depths, depth image, and road flood map. The outputs menu is only accessible after the algorithms have completed and the outputs have been generated. The user also has the option to save a GeoTIFF copy of any of the outputs they are currently viewing by pressing the "Save Current Image" button.

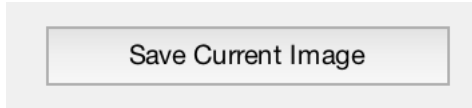


Figure 19: The Save Current Image Button

The flood map selection will display the flood image with the detected flood pixels shown in teal. The user can zoom, pan, and save the image.

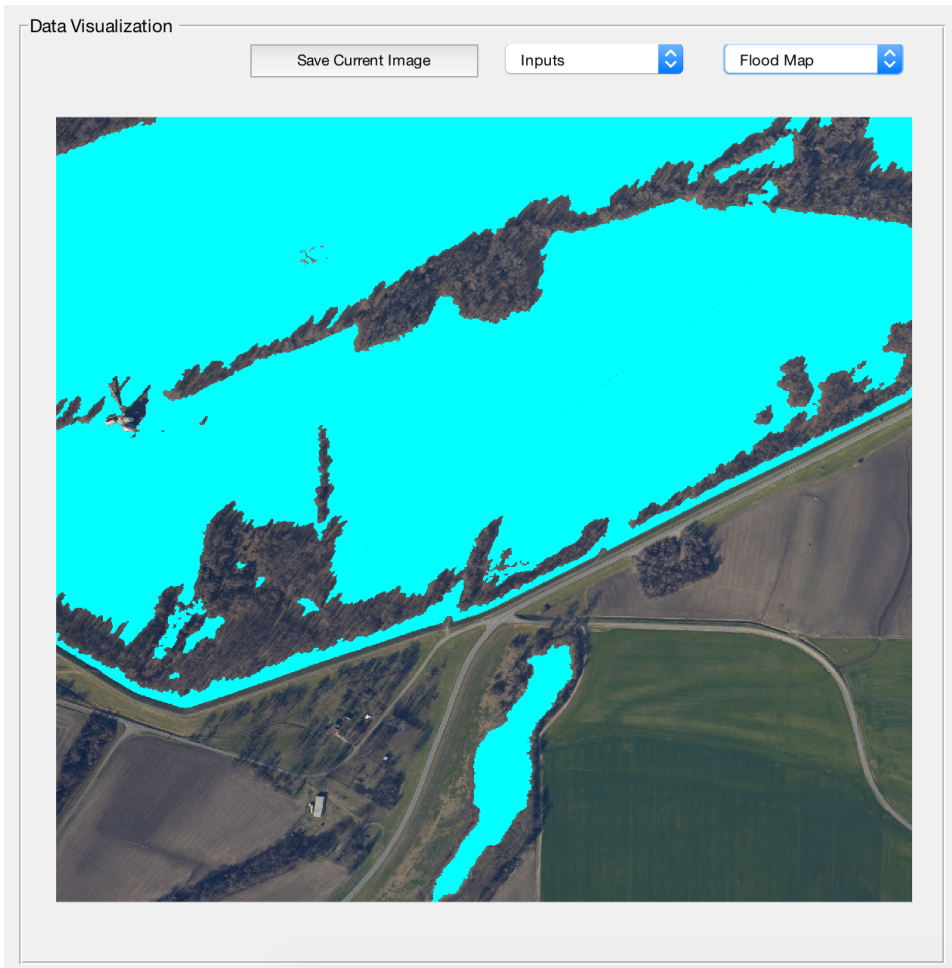


Figure 20: Visualizing the flood map image

The road depths selection will display the depth map for road pixels along with a colorbar

to the right indicating the depth scale (in the same units as the input DEM). The user can zoom, pan, and save the image.

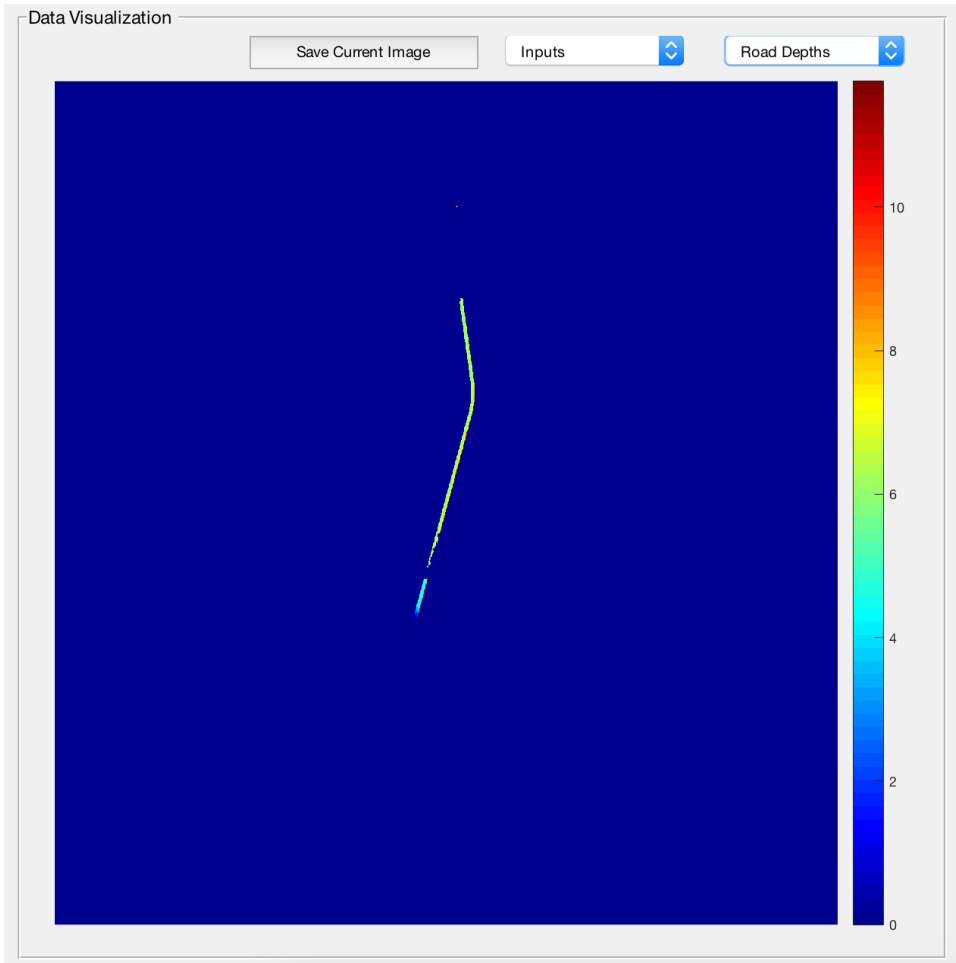


Figure 21: Visualizing the road depth image

The depth map selection will display the depth map for all pixels along with a colorbar to the right indicating the depth scale (in the same units as the input DEM). The user can zoom, pan, and save the image.

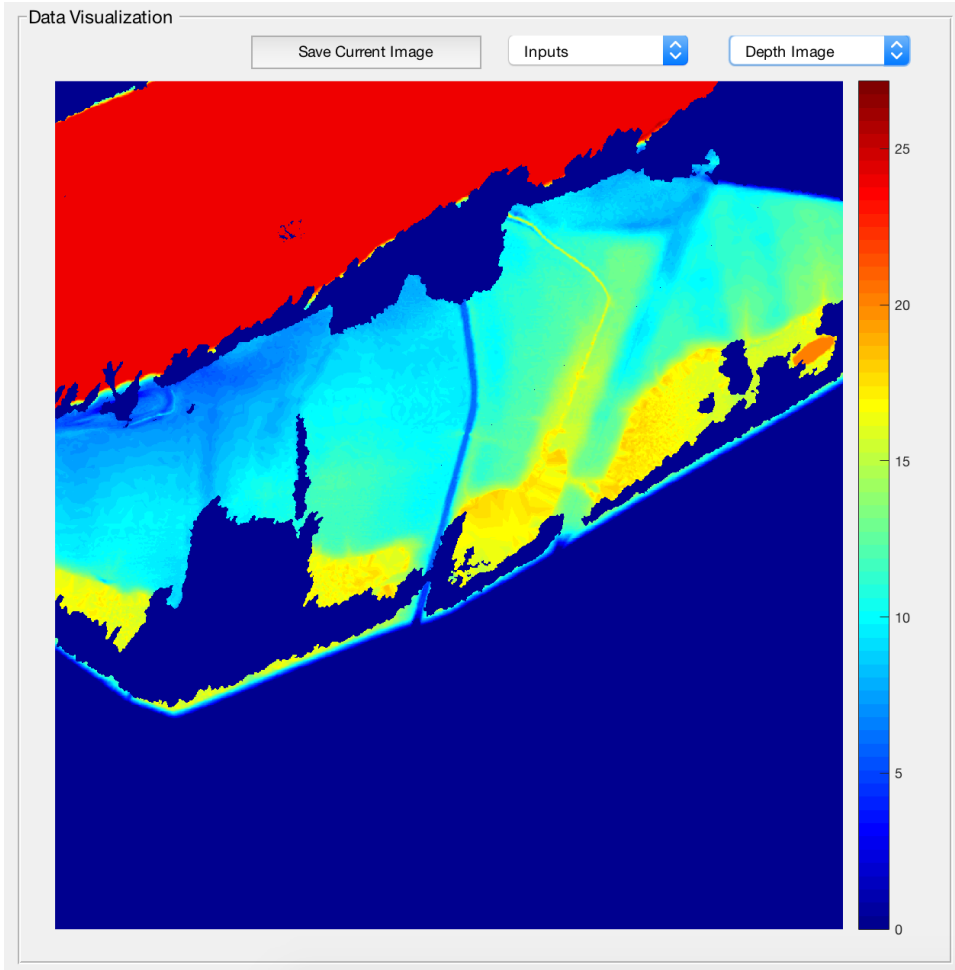


Figure 22: Visualizing the depth map image

The road flood map selection will display the flood image with the detected roadway flood pixels shown in teal. The user can zoom, pan, and save the image.

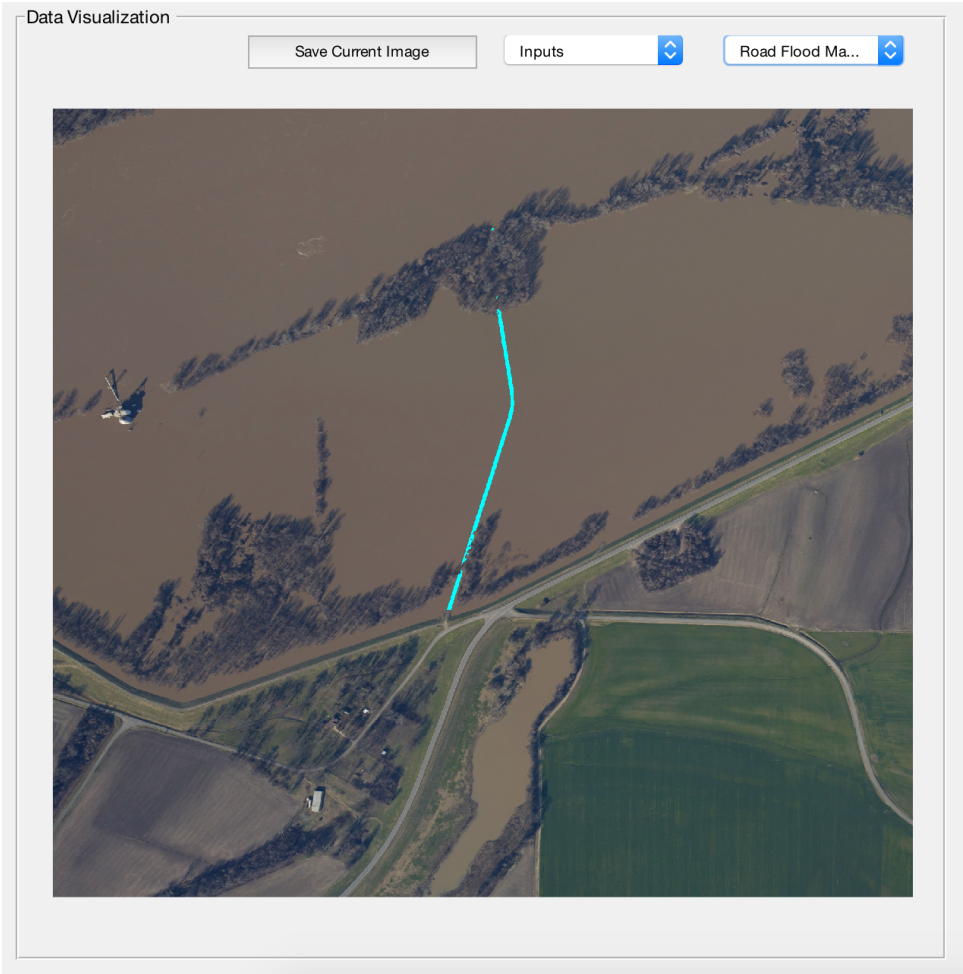


Figure 23: Visualizing the flood map image

5.4 The Run Settings Panel

The run settings panel (the box labeled 4 in Figure 8) is where the user can change the algorithm parameters prior to running (optional), the run mode, and then actually run the algorithms. The run settings panel is divided into four subsections: (1) algorithm parameters (2) the run method panel, (3) the restore defaults button, and (4) the run button.

5.4.1 The Algorithm Parameters

The algorithm parameters section is used to control some of the settings for the flood detection algorithm. The "QMF Threshold:" parameter controls the threshold for the result of the quadratic matched filter operation applied to the image. This setting determines how strict the algorithm is in classifying a pixel as a flood pixel. A higher value means the algorithm is more strict (less pixels will be detected; less false positives, but potentially more false negatives) and a lower value means the algorithm is less strict (more pixels will be detected; less false negatives, but potentially more false positives).

The "Open Radius [px]:" parameter controls the radius (in pixels) of the disk structuring element used for the morphological opening on the detected flood pixels. The opening is performed to clean the image, removing small connections between adjacent pixels unlikely to be flooding. A higher value for the open radius will remove more flood pixels while a smaller value for the open radius will remove less pixels.

The "Open Size [px]:" parameter controls the threshold (in pixels) for removing small connected components from the flood map. Components with less connected pixels than the threshold will be removed (e.g. components with less than 500 touching pixels will be removed if the default value is used). A higher open size will remove more components while a lower open size will remove more components.

The "# Selected Pixels:" text display will update as flood pixels are manually selected (if in pixel selection mode, otherwise it will read N/A).

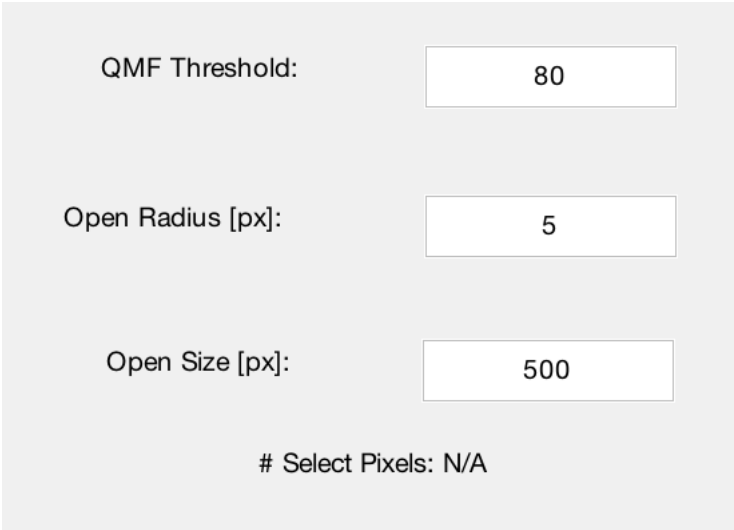


Figure 24: The algorithm parameters can be edited using these three edit boxes

5.4.2 The Run Method Panel

The Run Method panel is used to determine the type and extent of the algorithm used. The first choice the user makes is run method: automatic or pixel selection. This choice refers to how the algorithm finds the initial seed flood pixel from which the others will be detected. If automatic mode is chosen, the algorithm will use image processing to attempt to automatically detect a flood pixel. If pixel selection mode is chosen, the user can utilize the point picking tool to manually pick one or more flood pixels in the image. The pixel selection mode is recommended as it guarantees a flood pixel is chosen. Choosing pixel selection mode will also activate the "# Selected Pixels:" text display.

The second choice the user must make is to what extent the algorithm will run; the three run choices are in the Action panel. "Detect Flood Only" performs flood detection on a flood image and produces a flood map, but no depth estimates or shapefile output are conducted. The flood map option in the Outputs menu will become available after the algorithm completes. This mode requires only a flood image to be loaded.

"Detect Flood (in Roads)" performs flood detection on a flood image and also finds the flood pixels that are located in the road, but no depth estimates or shapefile output are conducted. The flood map and road flood map options in the Outputs menu will become available after the algorithm completes. This mode requires a flood image and a road network shapefile to be loaded.

"Detect Flood and Estimate Depth" runs the full algorithm, it: performs flood detection, water depth estimation, and outputs a shapefile to the selected output directory. All options in the Outputs menu will become available after the algorithm completes. This mode requires a flood image, road network shapefile, DEM image, and an output directory.

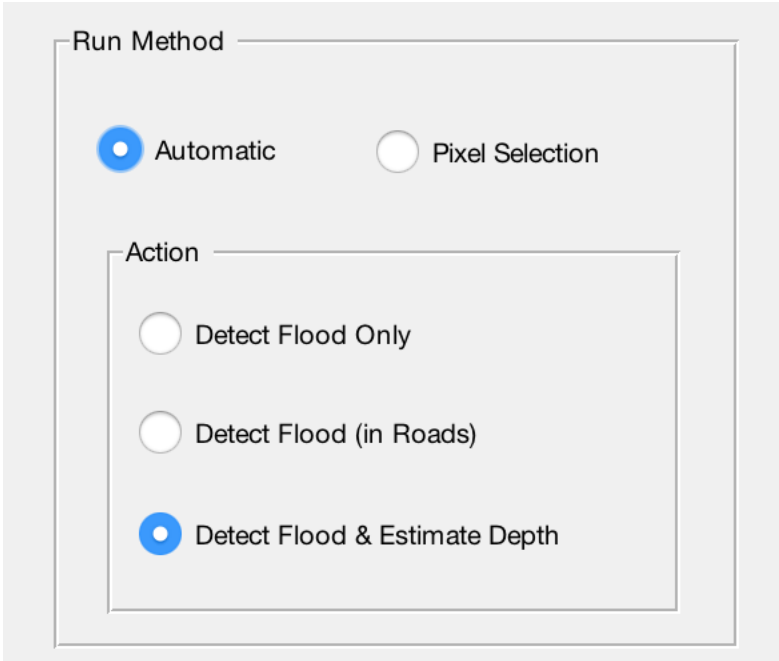


Figure 25: The run settings panel dictates how the algorithm executes

5.4.3 The Restore Defaults Button

The restore defaults button restores the QMF threshold to 80, the radius size to 5 pixels, the open size to 500 pixels, and the run mode to automatic. Any selected pixels will be deleted.

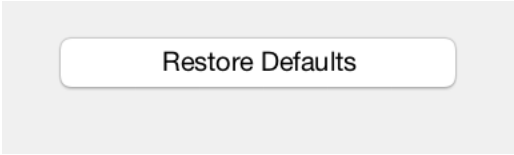


Figure 26: The Restore Defaults Button

5.4.4 The Run Button

The run button will execute the algorithm using the currently selected settings and parameters. The outputs menu in the data visualization panel will become clickable and the outputs that were created (based on the run choice the user selected) will become viewable.

If “Detect Flood and Estimate Depth” was selected, a shapefile will be created in the selected output directory. The shapefile will be named after the input flood image filename. For example, if the flood image is name “flood1.tif”, the output shapefile will be named “flood1_output.shp”.

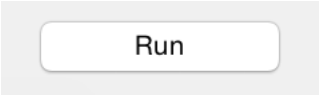


Figure 27: The Run Button

5.5 Running an Example

This section will run through an example of single file mode using real data. For this example, a folder called “flood_files” will be used as the source of input data (see Figure 47).

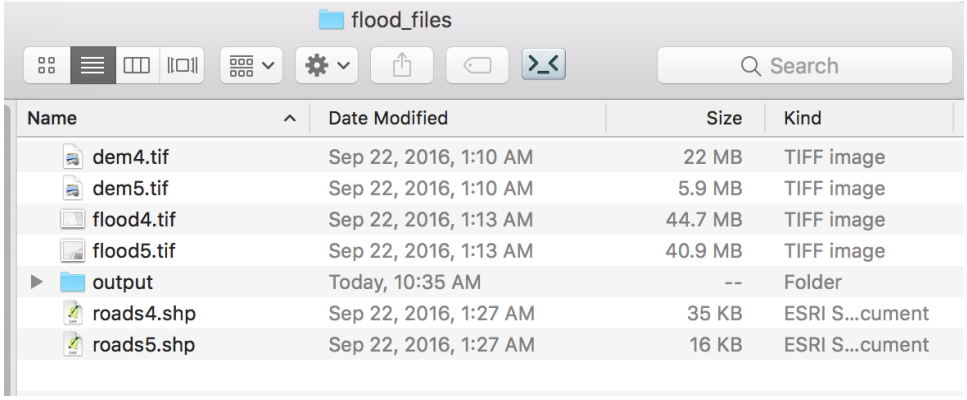


Figure 28: Input file folder for example

In this example, the files “flood4.tif”, “roads4.tif”, and “dem4.tif” will be used as inputs and the output shapefile will be stored in “flood_files/output”.

1. Click Single File Mode in the main menu.
2. Click the browse button in the Image File section and select “flood4.tif”. Hit the load button.
3. Click the browse button in the SHP section and select “roads4.shp”. Hit the load button.

4. Click the browse button in the DEM section and select “dem4.tif”. Hit the load button
5. Click the browse button in the Output Directory section and select “flood_files/output”.
6. Go to the inputs menu in the data visualization panel and select “Image”.
7. Change the run mode to pixel selection, change the action to “Detect Flood & Estimate Depth”, and use the point picking tool to pick a flood pixel on the image. The screen should now look like it does in Figure 29.

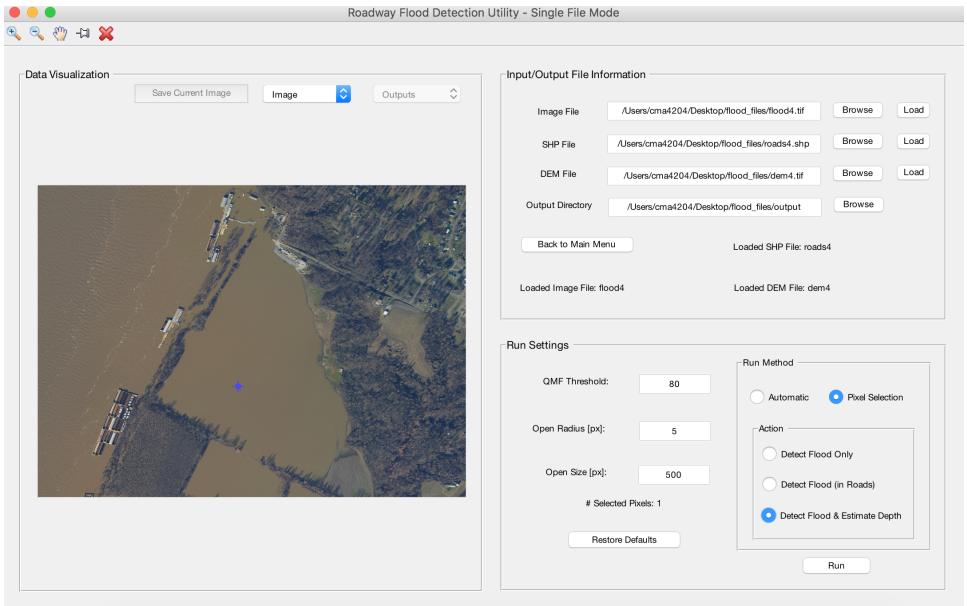


Figure 29: What the application should look like after loading the input data and selecting a flood pixel

8. The default parameters are perfectly fine for these data, so click the run button. A dialog box will pop up to let you know the algorithm is running.
9. Change the data visualization to the flood map image by changing the output menu to “Flood Map”. The algorithm was able to accurately detect flood pixels (Figure 30).

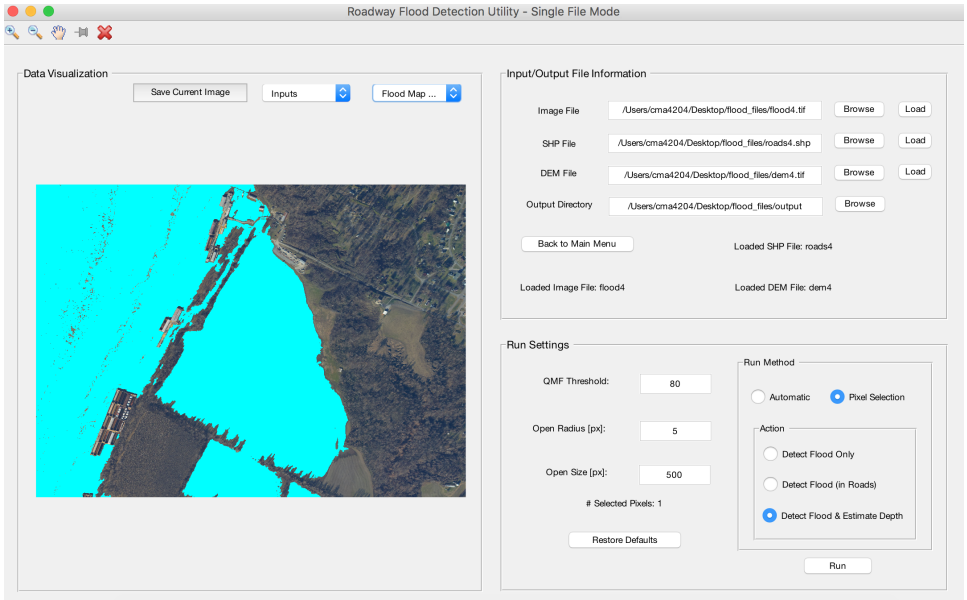


Figure 30: What the application should look like when visualizing the flood map results

10. Next, look in the output directory. The flood shapefile should be there.

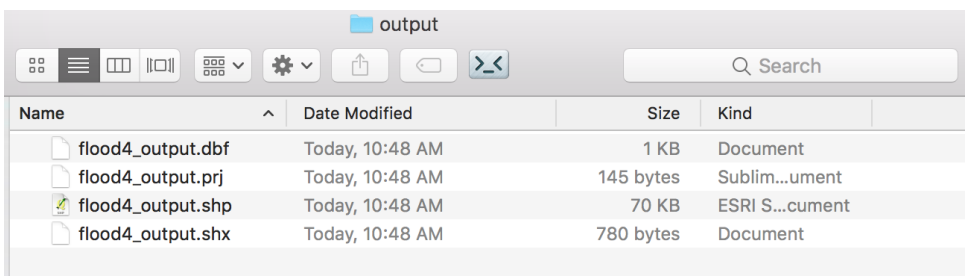


Figure 31: The flood shapefile appears in the output directory after a successful run

11. The output shapefile is in the output directory as expected and can be viewed in any GIS program. Congratulations, the example is complete.

6 The Batch File Mode Interface

The batch file mode interface pops up when selecting batch file mode from the main menu. This interface contains all of the controls and features needed to perform flood detection,

depth estimation, and shapefile output for all flood image/road shapefile/DEM image triplets in a single directory. The interface is divided into four sections: (1) the toolbar, (2) the input/output file panel, (3) the data visualization panel, and (4) the run settings panel. The interface with the four sections labeled is displayed in Figure 32. Each of these four sections and their components will be described in separate subsections.

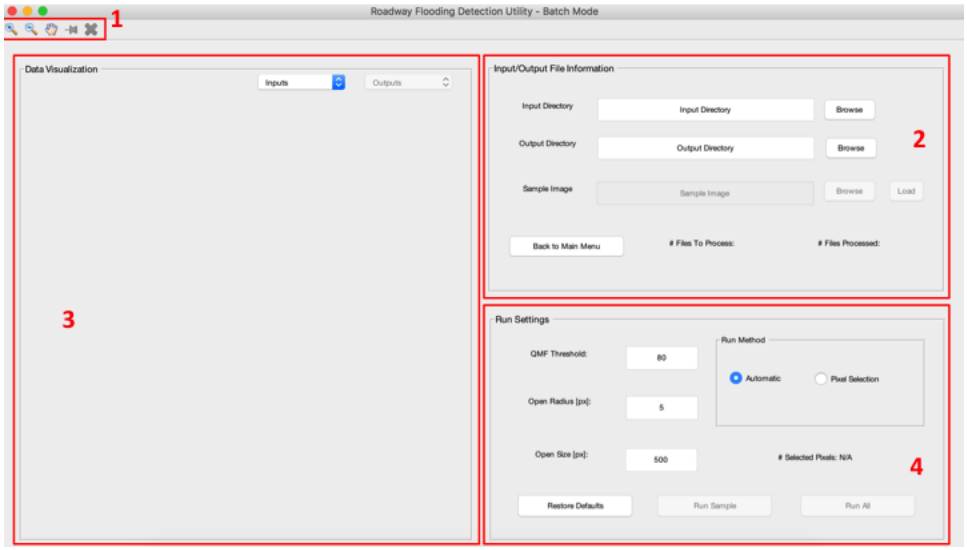
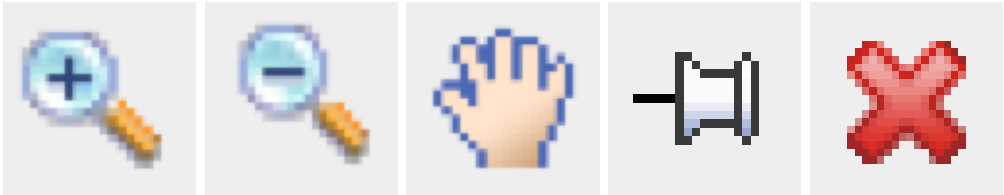


Figure 32: The Batch File Mode Interface

6.1 The Toolbar

The toolbar (the box labeled 1 in Figure 32) contains five tools thats can be used when viewing flood images or flood maps. Different input and outputs can be visualized by selecting one of the options from the input or output menu. Once a feature has appeared in the visualization panel, the icons on the toolbar let you do three things: zoom in, zoom out, and pan. If the feature is the flood image, two more tools are also available: pick a point and delete a point. The five tools will all be described in their own subsections.



(a) (b) (c) (d) (e)

Figure 33: The five tools of the toolbar: (a) zoom in (b) zoom out (c) pan (d) pick a point (e) delete a point

6.1.1 Zoom In

This tool allows the user to zoom in when viewing a flood image. Sometimes images are very large and it is hard to see what is going on when viewing the whole image. In these cases, use the zoom in tool to view a specific region. When the tool is selected, click the image to zoom into a specific point or click and drag a box to zoom into a whole region.

6.1.2 Zoom Out

This tool allows the user to zoom out when viewing a flood image. This tool only works after the screen has already been zoomed in. When the tool is selected, click the image anywhere to zoom out.

6.1.3 Pan

This tool allows the user to navigate the display after it has been zoomed in. When the tool is selected, click the image and drag.

6.1.4 Pick a Point

This tool allows a user to manually select flood pixels in a flood image. The tool becomes clickable when the data visualization panel is showing the flood image and the "pixel selection" button has been clicked in the Run Settings panel. To select a pixel, simply click the pick a point tool and then click a flood pixel in the image. Multiple pixels can be selected by repeating this process (the tool must be clicked again to select a new point).

Every time a pixel is selected, the "Selected Pixels" counter in the Run Settings panel will increase by one.

6.1.5 Delete a Point

This tool allows a user to delete selected flood pixels in a flood image. The tool becomes clickable as soon as a single flood pixels has been selected. To delete a point, simply click the tool. Clicking the tool will only delete the most recently selected point. The tool can be clicked as long as there are still points present (the points will be visible on the image screen). Every time a point is deleted, the "Selected Pixels" counter in the Run Settings panel will decrease by one.

6.2 The Input/Output File Panel

The input/output file panel (the box labeled 2 in Figure 32) is where the user selects and loads the input directory that contains the flood images, road network shapefiles, and DEM images, as well as the output directory in which the output shapefiles will go. Additionally, the user can choose to return to the main menu from this panel. The input images and shapefiles can be in either latitude/longitude or UTM coordinates. For the sake of this manual, the input/output panel can be divided into four subsections: (1) the input directory section, (2) the output directory section, (3) the back button, and (4) the text display.

6.2.1 The Input Directory Section

The Input Directory section contains all of the tools needed to select a directory in which the flood image/road network shapefile/DEM image triplets are stored. The text edit bar allows the user to manually enter a directory by typing or copying and pasting the path. If the entered path doesn't lead to a valid directory, an error will appear. The browse button lets the user navigate to an input directory At any point, the user can select a new input directory. Batch file mode requires corresponding flood images (*.tif/*.tiff), road network shapefiles (*.shp), and DEM images (*.tif/*.tiff), to be all numbered the same. For example: "flood1.tif", "roads1.tif", and "dem1.tif" or "nyc_f2.tif", "nyc_r2.tif", and "nyc_d2.tif". The naming requirement is what allows the algorithm to automatically choose and process the correct sets of files. When the input directory is selected, the "# Files to Process:" display will update with the number of detected file triplets and the "# Files Processed:" display will default to zero.

Figure 34: The Input Directory section

6.2.2 The Output Directory Section

The Output Directory section contains all of the tools needed to select a directory in which the output shapefile will be saved. The text edit bar allows the user to manually enter a directory by typing or copying and pasting the path. If the entered path doesn't lead to a valid directory, an error will appear. The browse button lets the user navigate to an output directory At any point, the user can select a new output directory.

Figure 35: The Output Directory section

6.2.3 The Back Button

The back button serves one simple purpose - it brings the user back to the main menu. The back button should only be used when the user wants to switch running modes (from single to batch or vice versa). All currently loaded data will be cleared when hitting the back button (except for the saved shapefile, it will remain in the selected output directory).

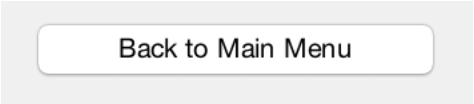


Figure 36: The Back Button

6.2.4 The Text Display

The text display exists mainly so that the user can see how many sets of file triplets (flood image, road network, DEM image) were detected in the input directory and also to see how many have been processed so far.

The "# Files to Process:" display will update with the number of detected file triplets as soon a valid input directory is selected.

The "# Files Processed:" display will default to zero as soon as a valid input directory is selected, and will increment by one each time a file is processed.

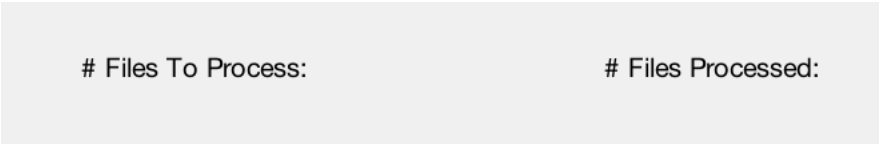


Figure 37: The Text Display section

6.3 The Data Visualization Panel

The data visualization panel (the box labeled 3 in Figure 32) is where the user can visualize the inputs and outputs (once generated). The data visualization panel doesn't become activated until the user chooses a input or output to view from the dropdown menus. The rest of this section will give examples of what the data visualization panel looks like for the various inputs/outputs.

6.3.1 The Inputs Dropdown Menu

The inputs dropdown menu lets the user visualize one of the flood images detected in the input directory. If the user tries to visualize an image file that hasn't been loaded yet, an error message will pop up. The purpose of viewing a flood image is to select pixels or adjust the algorithm parameters using a sample image before running match mode on a whole folder of images.

The image selection will display the flood image. The user can zoom, pan, and select/delete a point (if in pixel selection mode).

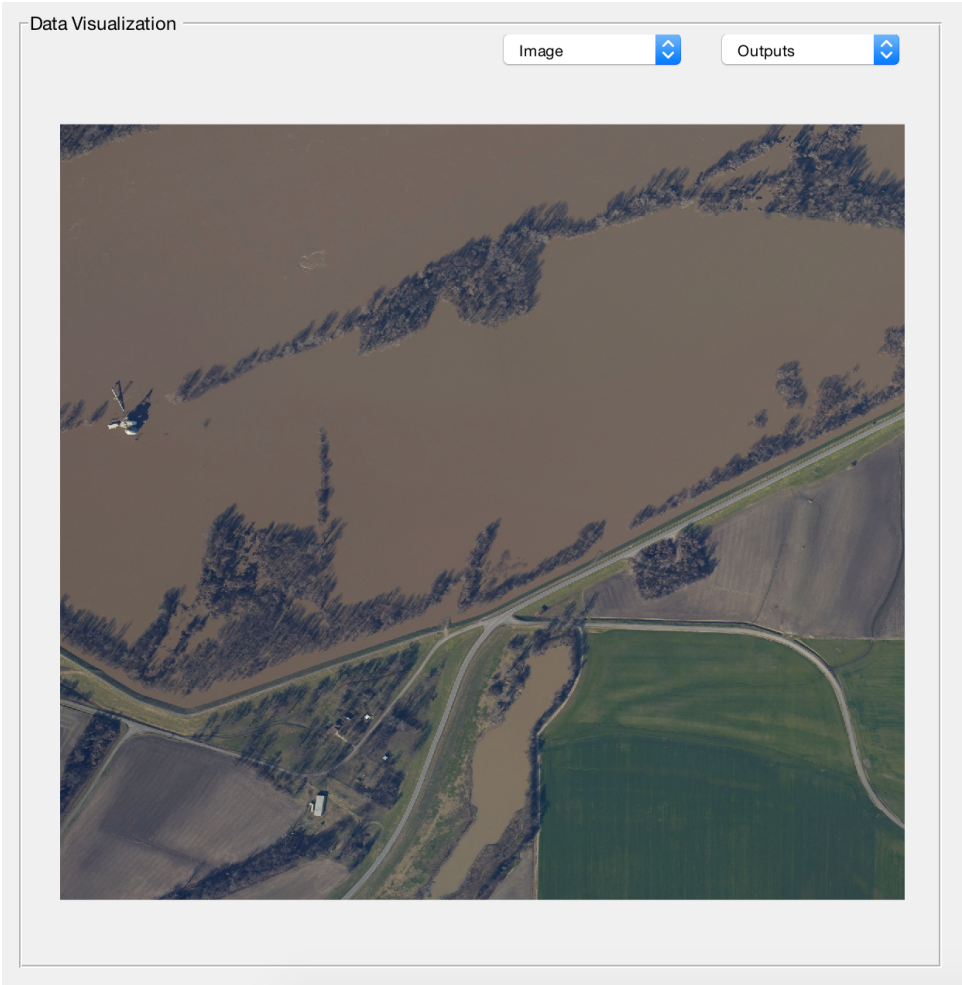


Figure 38: Visualizing the flood image

6.3.2 The Outputs Dropdown Menu

The outputs dropdown menu lets the user choose from four different choices: flood map, road depths, depth image, and road flood map. The outputs menu is only accessible after the algorithms have completed and the outputs have been generated. The outputs will correspond to the sample image (when using "Run Sample") or the last output (when using "Run").

The flood map selection will display the flood image with the detected flood pixels shown in teal. The user can zoom and pan.

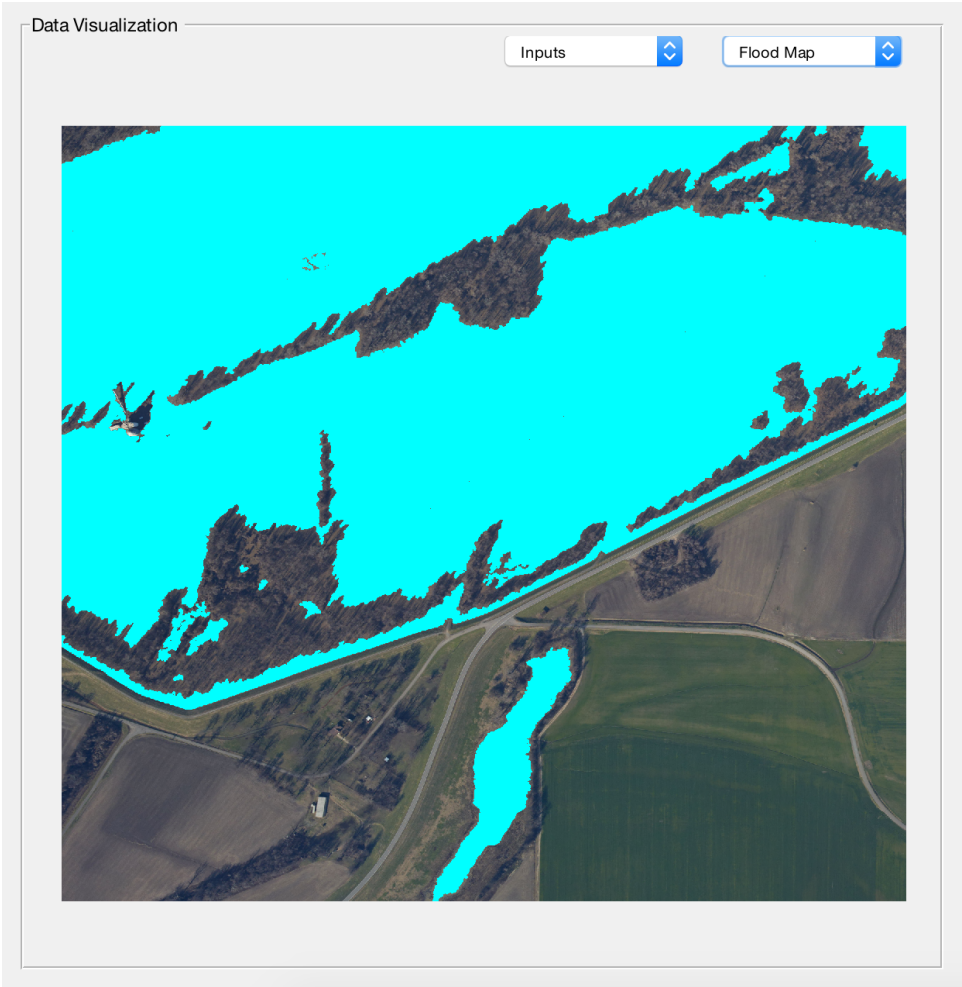


Figure 39: Visualizing the flood map image

The road depths selection will display the depth map for road pixels along with a colorbar to the right indicating the depth scale (in the same units as the input DEM). The user can zoom and pan.

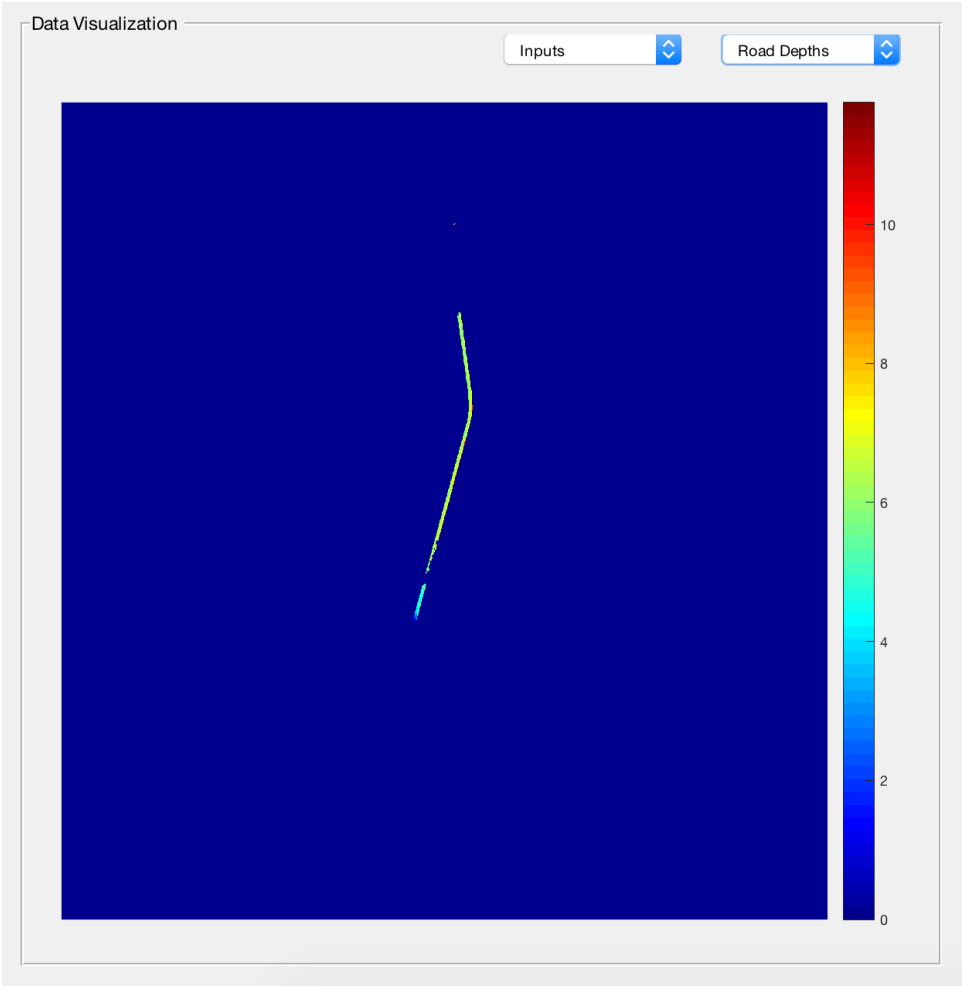


Figure 40: Visualizing the road depth image

The depth map selection will display the depth map for all pixels along with a colorbar to the right indicating the depth scale (in the same units as the input DEM). The user can zoom and pan.

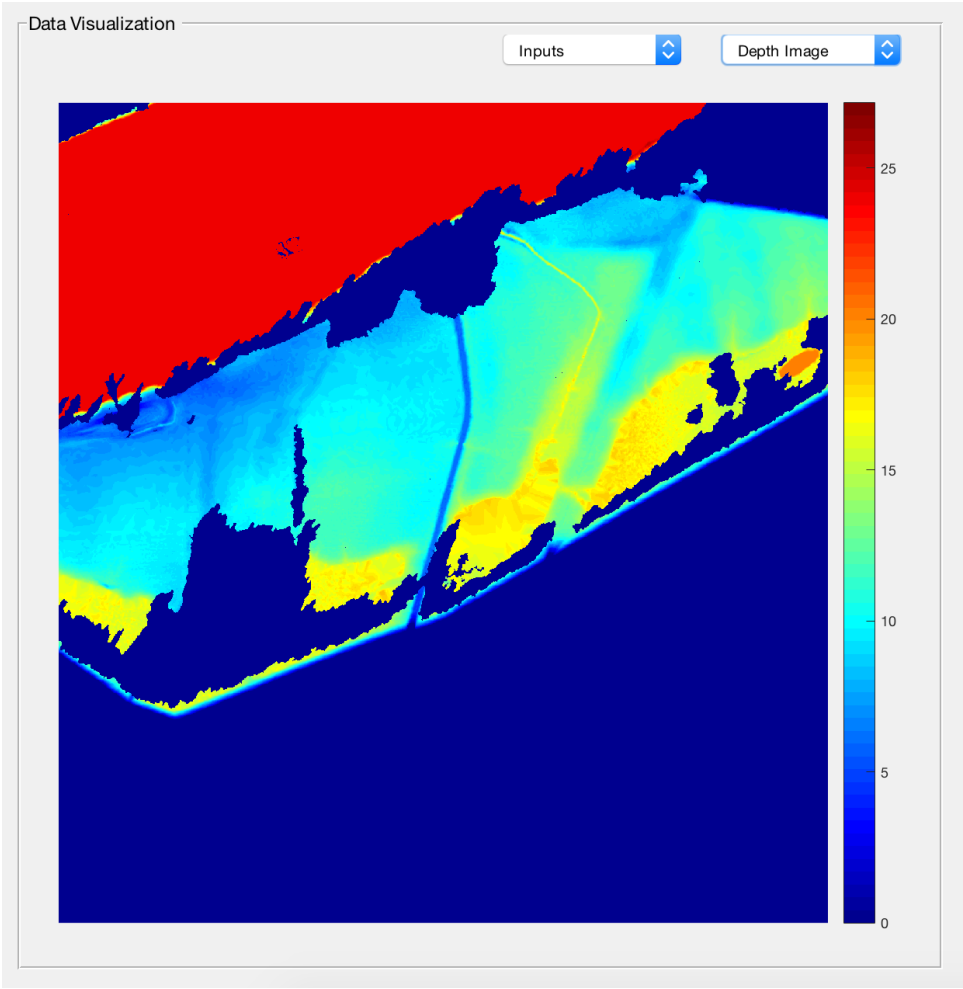


Figure 41: Visualizing the depth map image

The road flood map selection will display the flood image with the detected roadway flood pixels shown in teal. The user can zoom and pan.

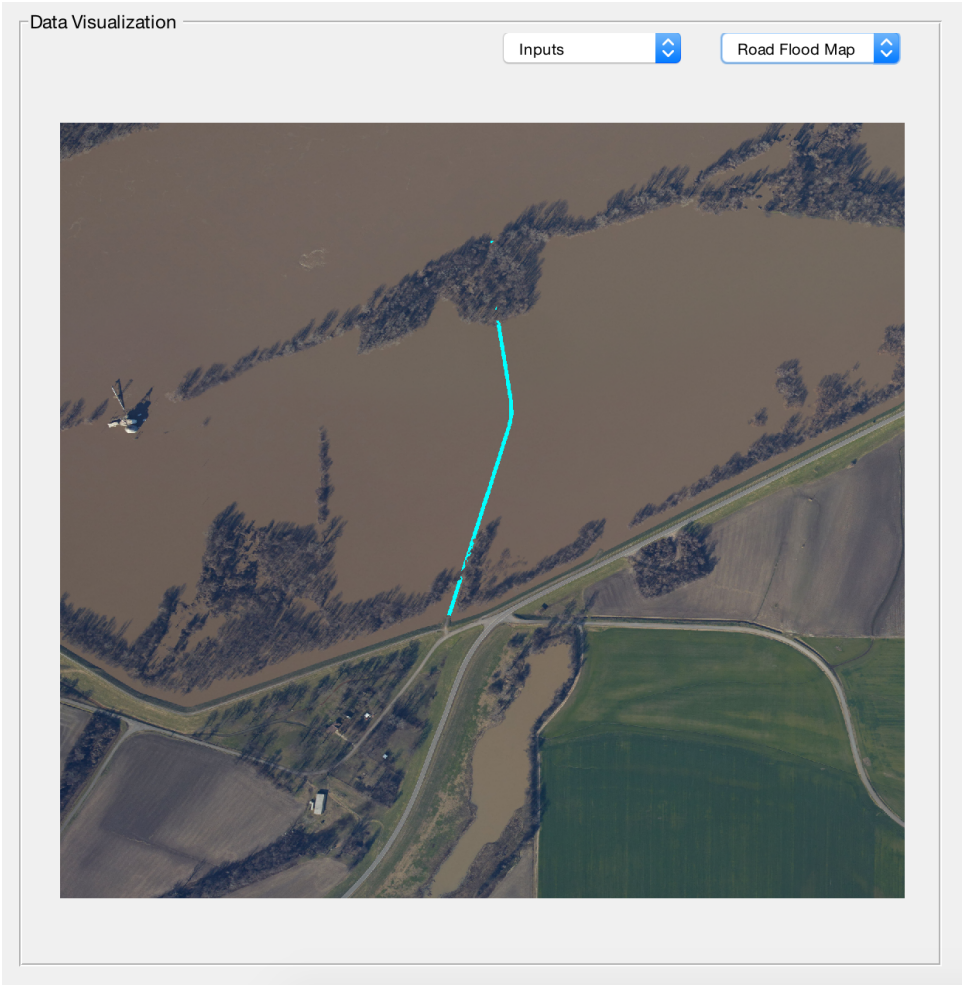


Figure 42: Visualizing the flood map image

6.4 The Run Settings Panel

The run settings panel (the box labeled 4 in Figure 32) is where the user can change the algorithm parameters prior to running (optional), the run mode, and then actually run the algorithms. The run settings panel is divided into four subsections: (1) algorithm parameters (2) the run method panel, (3) the restore defaults button, and (4) the run buttons.

6.4.1 The Algorithm Parameters

The algorithm parameters section is used to control some of the settings for the flood detection algorithm. The "QMF Threshold:" parameter controls the threshold for the result of the quadratic matched filter operation applied to the image. This setting determines how strict the algorithm is in classifying a pixel as a flood pixel. A higher value means the algorithm is more strict (less pixels will be detected; less false positives, but potentially more false negatives) and a lower value means the algorithm is less strict (more pixels will be detected; less false negatives, but potentially more false positives).

The "Open Radius [px]:" parameter controls the radius (in pixels) of the disk structuring element used for the morphological opening on the detected flood pixels. The opening is performed to clean the image, removing small connections between adjacent pixels unlikely to be flooding. A higher value for the open radius will remove more flood pixels while a smaller value for the open radius will remove less pixels.

The "Open Size [px]:" parameter controls the threshold (in pixels) for removing small connected components from the flood map. Components with less connected pixels than the threshold will be removed (e.g. components with less than 500 touching pixels will be removed if the default value is used). A higher open size will remove more components while a lower open size will remove more components.

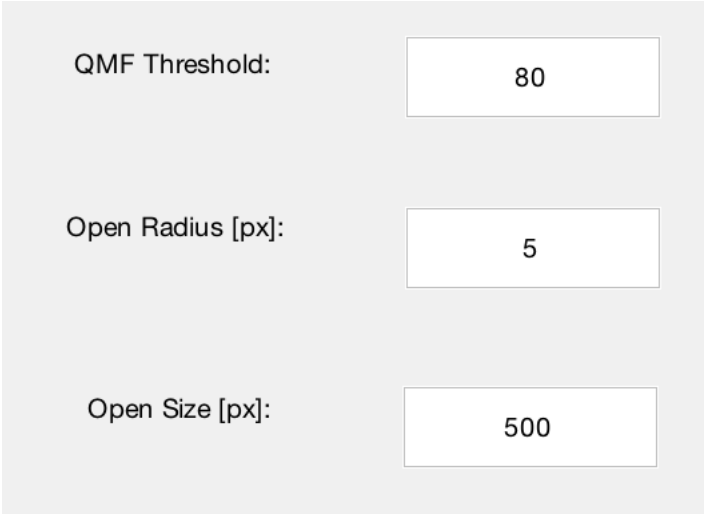


Figure 43: The algorithm parameters can be edited using these three edit boxes

6.4.2 The Run Method Panel

The Run Method panel is used to determine the flood detection mode. The user must decide a run method: automatic or pixel selection. This choice refers to how the algorithm finds the initial seed flood pixel from which the others will be detected. If automatic mode is chosen, the algorithm will use image processing to attempt to automatically detect a flood pixel. If pixel selection mode is chosen, the user can utilize the point picking tool to manually pick one or more flood pixels in the image. The pixel selection mode is recommended as it guarantees a flood pixel is chosen. Choosing pixel selection mode will also activate the "# Selected Pixels:" text display.

The "# Selected Pixels:" text display indicates how many flood pixels have been selected.

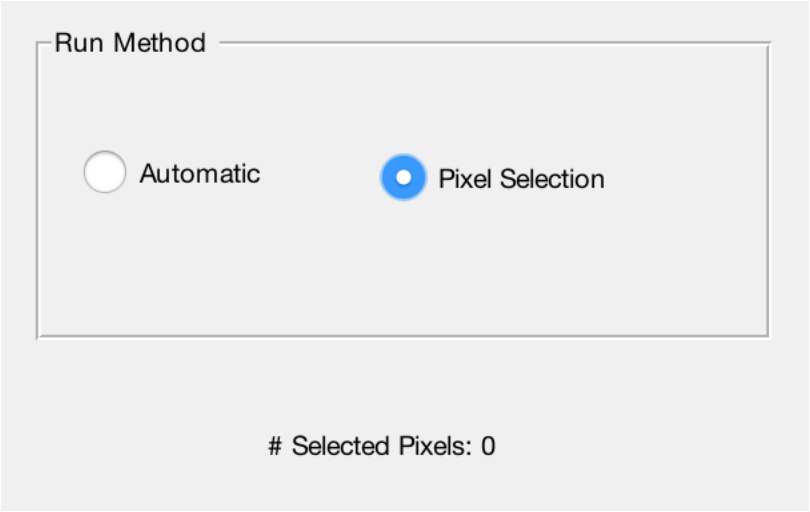


Figure 44: The run settings panel dictates how the algorithm executes

6.4.3 The Restore Defaults Button

The restore defaults button restores the QMF threshold to 80, the radius size to 5 pixels, the open size to 500 pixels, and the run mode to automatic. Any selected pixels will be deleted.

Figure 45: The Restore Defaults Button

6.4.4 The Run Buttons

There are two run buttons in batch file mode: Run Sample and Run All.

The Run Sample button becomes clickable once a sample image is loaded. The purpose of the run sample button is to allow the user to choose parameters and select flood pixels and test them to see how they work on a sample image. If those parameters work good on the sample image (and the sample image is similar to other images in the directory) then those settings should be used for all of the images.

The Run All button becomes clickable once an input and output directory have been selected. The Run All button runs the entire algorithm on every image/road/DEM triplet in the input directory, outputting a shapefile for each set in the output directory. A progress bar will appear indicating how much longer it will take to finish running on all of the files.

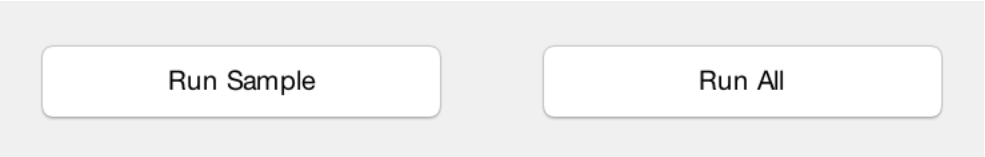


Figure 46: The Run Buttons

6.5 Running an Example

This section will run through an example of batch file mode using real data. For this example, a folder called "flood_files" will be used as the source of input data (see Figure 47).

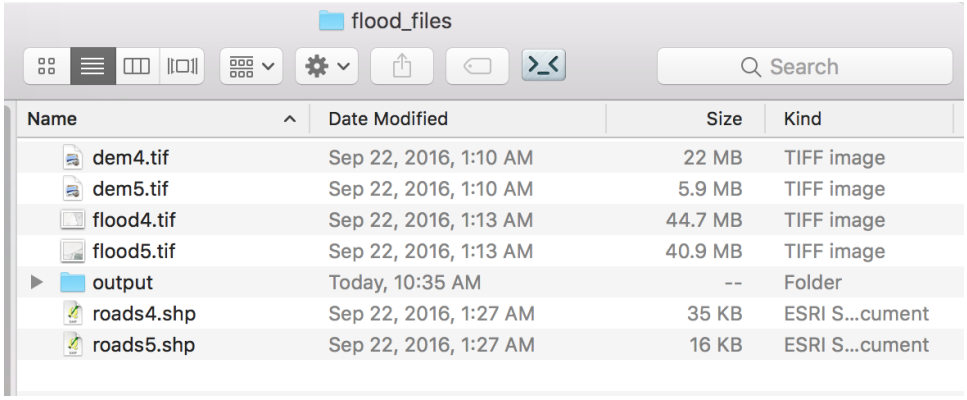


Figure 47: Input file folder for example

In this example, the files "flood4.tif", "roads4.tif", and "dem4.tif" as well as 'flood5.tif", "roads5.tif", and "dem5.tif" will be used as inputs and the output shapefiles will be stored in "flood_files/output".

1. Click Batch File Mode in the main menu.
2. Click the browse button in the Input Directory section and select "flood_files". Notice how the "# Files to Process:" counter updates to 2.
3. Click the browse button in the Output Directory section and select "flood_files/output".
4. Click the browse button in the Sample Image section and select "flood_files/flood4.tif" and then click load. The flood image will appear in the data visualization tab.
5. For this example we are going to pick a flood pixel manually. Change the run method to pixel selection, click the point picking tool, and then select a few flood pixels. The application should now look like it does in Figure 48.

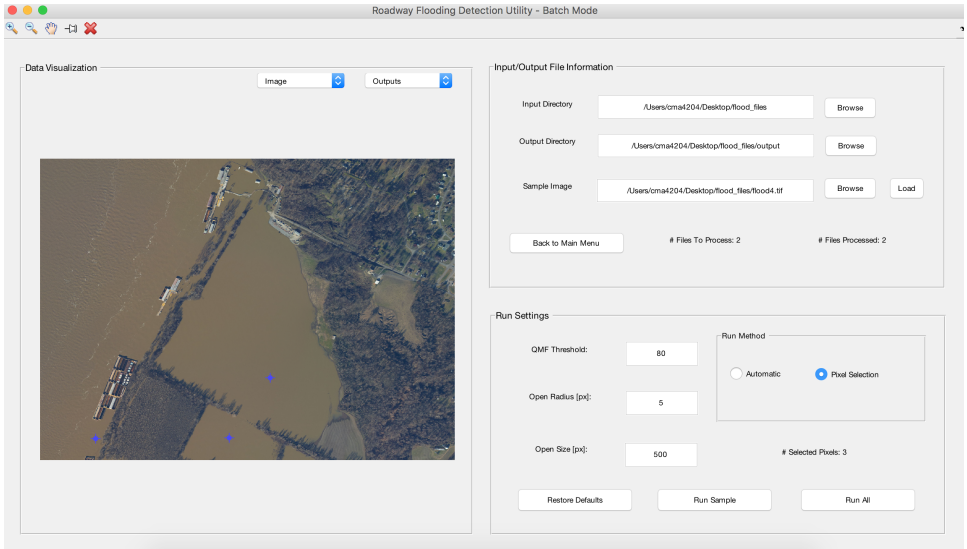


Figure 48: What the application should look like after selecting some flood pixels in the sample flood image

6. Test the performance of the algorithm with the current settings by pressing Run Sample. Once flood detection finishes, visualize the results by selecting Flood Map from the outputs menu. The results look accurate (Figure 49), so now run the algorithm for all files by hitting Run All. A progress bar will appear that updates after each file is processed.

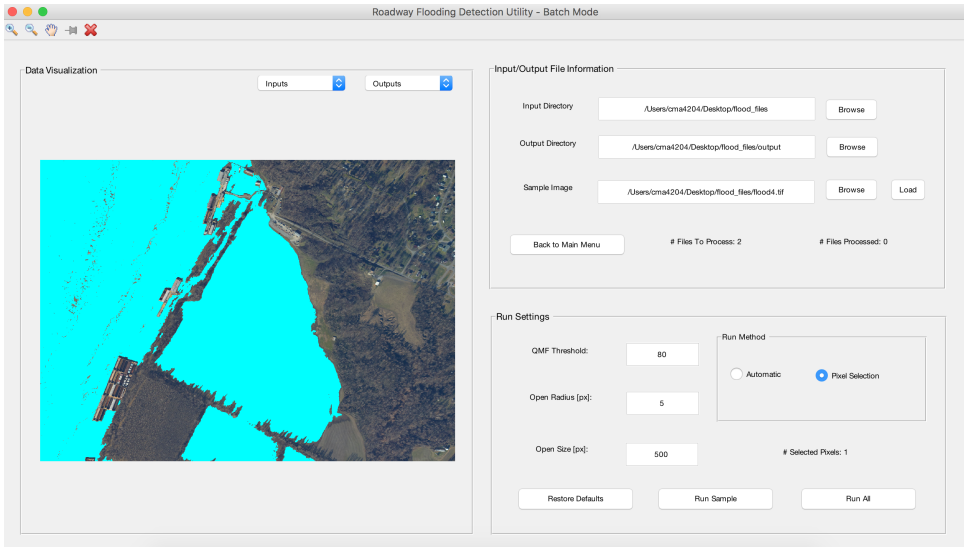


Figure 49: Visualizing the results from running flood detection on the sample

7. After the algorithm completes, visualize the results by selecting Road Flood Map from the outputs menu. This will show the resulting road flood map for the last file that was processed (in this case "flood5.tif"). See Figure 50.

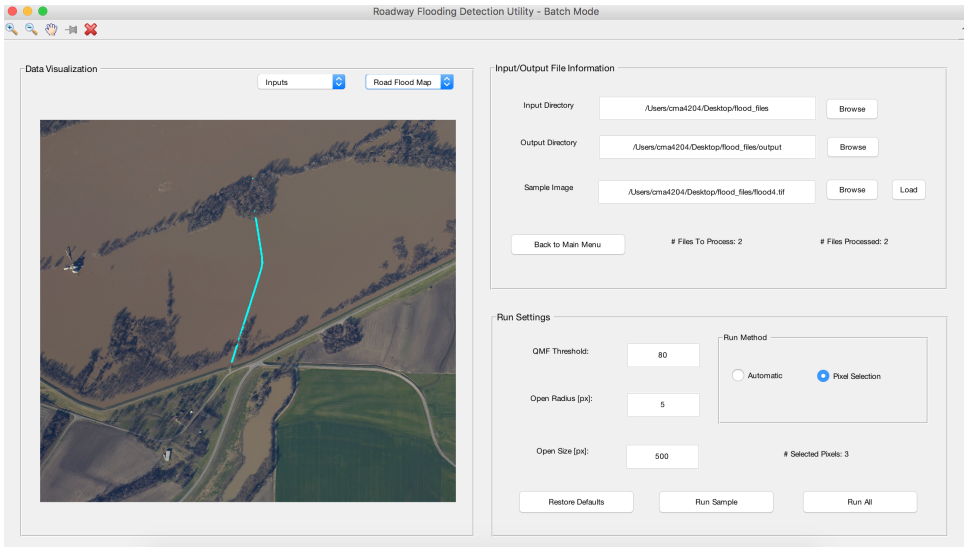


Figure 50: Visualizing the roadway flood map generated from running all files

8. Next, check to make sure that the output shapefiles were generated. Navigate to the output directory, "flood_files/output" (Figure 51).

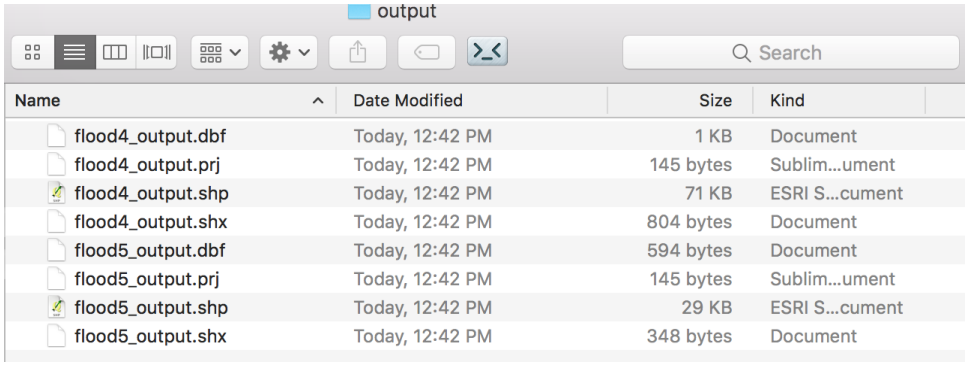


Figure 51: The shapefiles generated by the algorithm are in the output directory that was selected

9. The output shapefiles are in the directory. Congratulations, the algorithm ran successfully and the example is complete.