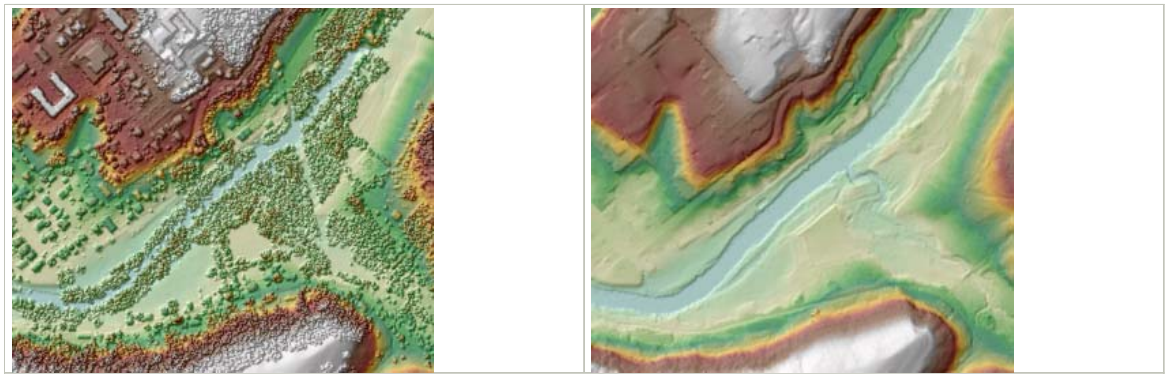
Creating raster DEMs and DSMs from large lidar point collections

ArcGIS 10

Summary

Raster, or gridded, elevation models are one of the most common GIS data types. They can be used in many ways for analysis and are easily shared. Lidar provides you with the opportunity to make high-quality elevation models of two distinct flavors: first return and ground. A first return surface includes tree canopy and buildings and is often referred to as a digital surface model (DSM). The ground, or bare earth, contains only the topography and is frequently called a digital elevation model (DEM).

These graphics show hillshaded representations of a first return surface, or DSM, on the left and a bare earth model, or DEM, on the right.



Coming up with a plan

Before beginning to create a raster from lidar, some basic factors need to be evaluated:

- Extent of lidar coverage
- Number of lidar points and point density
- Desired output raster resolution
- Extent of output raster(s)
- Format of output raster(s)

Consideration of these factors will help determine whether you will be producing one raster or a collection of rasters. Part of this process requires that you figure out how many rows and columns you're willing to have in one raster. This depends on what you intend to do with the raster in terms of analysis, display, and potential sharing or distribution of the data. The desire to work off one dataset for analysis can be in conflict with practical constraints associated with dataset size. Another thing to consider is the amount of lidar you have. Trying to process 10 billion lidar points as one dataset, while possible, is likely to prove unwieldy. In this situation you would want to make multiple rasters from this volume of lidar data, so consider splitting up the lidar processing as well. Not only does this keep individual datasets at reasonable sizes, it also keeps process duration on those datasets shorter. The longer a process takes to execute, the more likely something's going to go wrong (for example, power outage).

If you have determined you need to split up your data, the next question is how can this be accomplished. Will it be based on a regular grid system, political boundaries, or an anticipated application? Since lidar collections tend to have multiple uses, splitting them up based on a regular grid system or political divisions like county boundaries makes the most sense. An engineer can mosaic the different pieces he or she needs for an individual project. If the intended use is weighted heavily for one type of application, such as hydrology, then use divisions logical for the application. For example, in the case of hydrology, watershed boundaries are a good candidate.

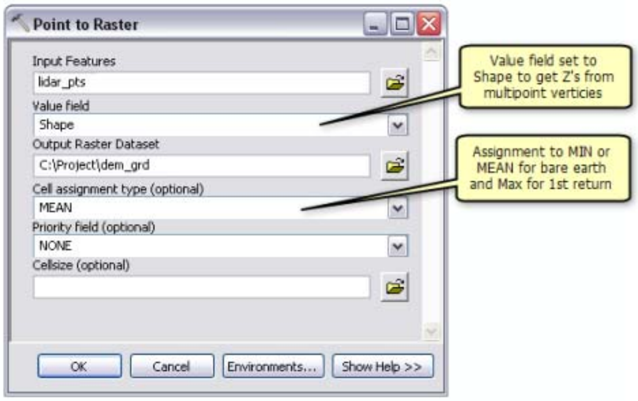
ArcGIS supports many raster formats, so you have a choice of what format to write to. This decision is best based on the intended use of the product. If it's to be shared with the general public you might think about distributing in either TIFF or JPEG format. For analysis using the ArcGIS platform, consider using the file-based geodatabase format.

The first step in getting from lidar points to raster is loading the points into a geodatabase. To load your lidar points into a multipoint feature class, use either the LAS To Multipoint or the ASCII 3D To Feature Class geoprocessing tool, depending on the source data format of the lidar data. Put the multipoint feature class in a feature dataset if you intend to build a terrain dataset from the lidar points. Although you have the choice between using LAS or ASCII format files, LAS is a more acceptable binary file format. LAS files contain more information and, being binary, can be read by the importer more efficiently. For more information on importing lidar source measurements into the geodatabase, see Data import and load tools.

Using the Point To Raster geoprocessing tool

If your only source of data is the lidar, you can use the Point To Raster geoprocessing tool to produce raster elevation models. The Point To Raster tool does not produce the highest quality result possible. Lidar tends to be so dense that for many applications the accuracy produced with the Point To Raster geoprocessing tool is good enough and the convenience and speed of this tool make it worthwhile.

If producing a bare earth raster surface, or DEM, use just the ground lidar points to produce the raster. Set the **Value field** parameter on the tool to Shape to use the z-values from the multipoint vertices. Also, set **Cell assignment type** to either **MIN** or **MEAN**. **MIN** will bias output heights to local lows, while **MEAN** is more of a general purpose option. To produce a first return surface, or DSM, use the first return lidar points with the **MAX** option of the tool since you want to bias the output to local highs.



Point to Raster

Input Features: lidar_pts

Value field: Shape

Output Raster Dataset: C:\Project\dem_grd

Cell assignment type (optional): MEAN

Priority field (optional): NONE

Cellsize (optional):

Value field set to Shape to get Z's from multipoint verticies

Assignment to MIN or MEAN for bare earth and Max for 1st return

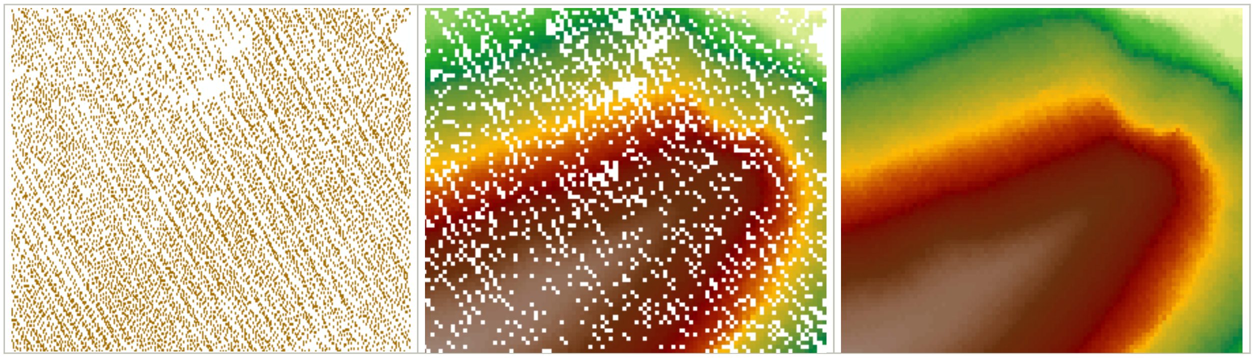
The Point To Raster geoprocessing tool produces gridded elevation models from lidar point sets.

While Point To Raster offers the easiest and fastest way to produce a raster from lidar, it has a significant drawback. You can end up with many NoData cells since it only returns values for cells that have one or more points in them. The problem is exacerbated when only using ground points to make a DEM because gaps in the data occur where there's vegetation and buildings obscuring the ground. To reduce the effect of NoData versus data cells, you can increase the output cellsize relative to the average point spacing. You can also reduce the number of NoData cells after execution of Point To Raster by using the sample below in the ***Python*** window if:

- The Spatial Analyst extension is on
- The output name of the last focal stats expression executed is 'outfocalmNN' (outfocalm03, outfocalm04, outfocalm05 etc.)
- The output (in this example, the output from Point To Raster is called point2ras) is a layer in the table of contents
- The valid path is: C:\data\

```
from arcpy.sa import *
outfocalm01 = FocalStatistics("point2ras", NbrRectangle(3, 3, "CELL")
outfocalm02 = FocalStatistics(outfocalm01, NbrRectangle(3, 3, "CELL")
#Repeat using output (temporary raster object) in the next processes until
all nodata is gone
out = Con(IsNull("point2ras"), outfocalmNN, "point2ras")
#Save the result to disk
out.save("C:\data\myfinalDEM")
```

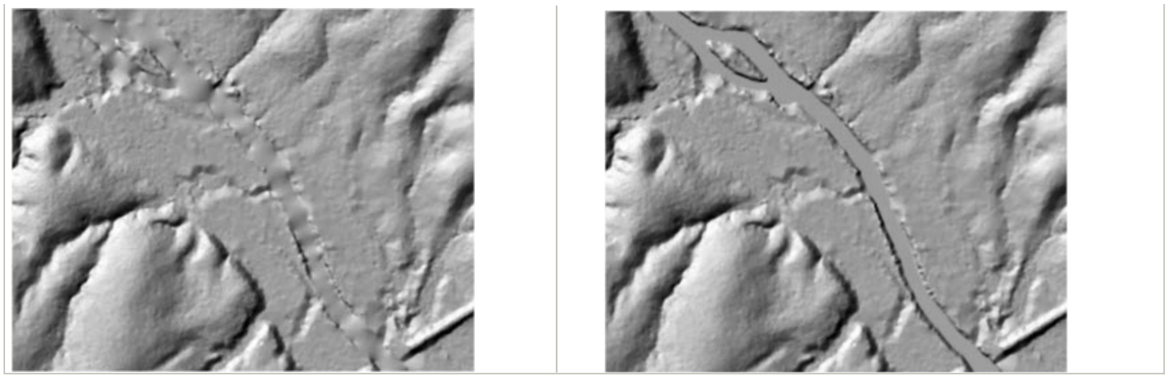
You can run the Python sample multiple times to fill in larger NoData areas, but it is not recommended to run it more than a couple times. It is better to just accept larger void areas as a consequence of using this approach.



Original points on left, output from Point To Raster in the middle (NoData cells colored white) in the middle, post-processed raster with NoData filled in on the right.

Using the terrain dataset to create a raster DEM

If you have photogrammetric breaklines to go along with your lidar, or need higher-quality results than can be produced with the Point To Raster tool, use the terrain dataset. For an overview of the terrain dataset, see What is a terrain dataset?



On the left is a surface made without breaklines along the river banks. The right-hand side has breakline enforcement. Breaklines are important for maintaining the definition of water-related features in the elevation model.

Breaklines are used to capture linear discontinuities in the surface. The most common types are edge of pavement, lake shorelines, single-line drains for small rivers, and double-line drains for large rivers. Sometimes breaklines are also collected to help define and sculpt the surface without necessarily representing discontinuities. Examples of these applications include contourlike form lines and the crests of rounded ridges.

Breaklines, while frequently used in bare earth models, tend to be detrimental when used with first return surfaces because they can be in conflict with the aboveground points. For example, breaklines capturing road edge of pavement can be coincident in x,y but different in z to points in the tree canopy overhanging the road. Because of this, consider excluding breaklines from your first return surface or at least those where you know there's potential conflict.

The most efficient means of organizing breaklines for use in a terrain dataset (see table below) is to separate them into different feature classes based on surface feature type (SFType). Surface feature types control how the features are enforced in the model and how the natural neighbor interpolator, used during rasterization, interprets the surface as it crosses over these features. A distinct break in slope will occur across "hard" features but not across "soft" features.

SFType for terrain datasets

Measurement type	Feature class type	SFType
Lidar points	3D multipoint feature class	Mass
Edge of pavement, single- and double-line drains for rivers, sharp ridgelines	3D line feature class	Hardline
Lakes, reservoirs	2D polygon feature class with z-value stored as attribute	Hardline or hardreplace
Eroded/Rounded ridgelines, contourlike form lines	3D line feature class	Softline
Study area boundary	2D polygon feature class;no z-value	Softclip

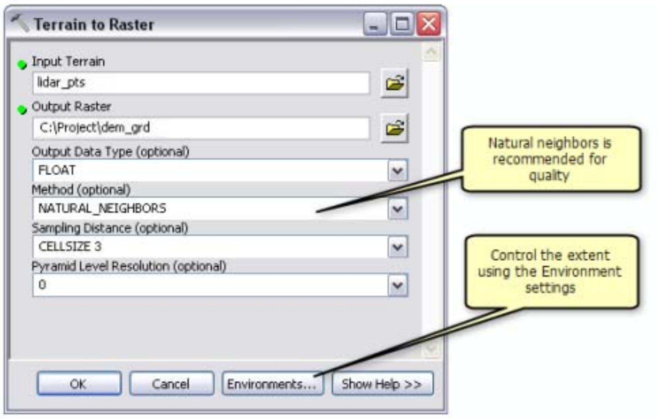
It is best for the sake of terrain performance to place all hardlines together in one feature class. It is understood this might not be possible, for example, if you need to keep road and water features separate. Keep in mind, the fewer feature classes used to define a terrain, the better.

Replace SFType is used to set everything inside a polygon at a constant height. It is used mostly for lakes when there's inadvertently other data inside them, such as lidar points, whose heights are not exactly the same as the shoreline and therefore prevent the water bodies from being flat. Replace SFType does incur more processing cost than normal hard- or softlines, so it is best to avoid using it in a terrain dataset. Ideally, there should not be lidar samples in water bodies (consider adding this as a stipulation in the contract with your data provider), but if there are, you can use the Delete Terrain Points geoprocessing tool to handle them after the terrain dataset is built. Otherwise, you can eliminate any offending points before building your terrain using the Erase Point geoprocessing tool.

If you will be producing both bare earth and first return surfaces using terrain datasets, load the lidar points into two different multipoint feature classes, a feature class for the ground points and a feature class for the aboveground points. The bare earth terrain is defined with a reference to only the ground points. The first return terrain dataset references the same ground point feature class as the bare earth terrain with the additional reference to the aboveground points. This means two different terrain datasets can reference the same feature class.

Starting with ArcGIS 9.3, terrain datasets can be pyramided using one of two point thinning filters: z-tolerance and window size. For DEM production, you can use either of these pyramid types. If you intend to rasterize from the full-resolution point set, use the window size filter for terrain construction because it is significantly faster. If you are willing to use thinned data for analysis, which is reasonable if the lidar data is oversampled for your needs, use the z-tolerance filter. While more time consuming, it is most appropriate because it provides an estimate of vertical accuracy of the thinned representation. For DSM production, use the window size filter with the MAX option.

Use the Terrain To Raster tool to produce your rasterized elevation model. This provides options for interpolation, output cellsize, and which pyramid level to use from the terrain dataset.



Input Terrain: lidar_pts

Output Raster: C:\Project\dem_grd

Output Data Type (optional): FLOAT

Method (optional): NATURAL_NEIGHBORS

Sampling Distance (optional): CELLSIZE 3

Pyramid Level Resolution (optional): 0

Natural neighbors is recommended for quality

Control the extent using the Environment settings

The Terrain To Raster tool produces gridded elevation models from terrain datasets.

For interpolation, the natural neighbors options is the best option. While not as fast as linear interpolation, it generally produces better results both in terms of aesthetics and accuracy. Set the output cellsize relative to the lidar point sample density. You will not gain any accuracy by using a cellsize that is substantially smaller than the average point spacing. Also, make sure to set the analysis extent, as set through the **Environments**, for the extraction of subsets where appropriate. The use of a snap raster can also be of use for the sake of alignment of raster outputs.

Overview steps to generate a raster DEM surface from lidar point data in ArcGIS are described below.

First, you will create a terrain dataset in ArcCatalog or the ***Catalog*** window inside your appliation, then use geoprocessing tools to convert this terrain dataset to a raster DEM.

1. Create a terrain dataset.

Steps:

1. Determine the source data and how that data will contribute to the terrain dataset.
   For more information on representing terrain source data, see Representing terrain source data in feature classes and Types of source data supported in terrain datasets.

   Note: All source data must have the same spatial reference to build a terrain dataset.

2. Create a file geodatabase in ArcCatalog or in the ***Catalog*** window. Right-click the folder where the terrain is to be built, point to **New**, then click **File Geodatabase** in the context menu.
3. Create a feature dataset. Right-click the file geodatabase, point to **New**, then click **Feature Dataset** in the context menu.
   For more information on correctly generating a feature dataset, see Creating a feature dataset.
4. Import the source measurements into feature classes. These feature classes must be produced inside the feature dataset created in step 3. For more information on how to import source data for a terrain, see Importing terrain dataset source measurements.
5. Build a terrain dataset in ArcCatalog or the ***Catalog*** window using the ***New Terrain*** wizard.
   To access the ***New Terrain*** wizard, right-click the feature dataset to display the menu, point to **New**, then click **Terrain**.
   For more information about using the ***New Terrain*** wizard, see Building a terrain dataset with the Terrain wizard.

2. Use the Terrain To Raster geoprocessing tool.

Steps:

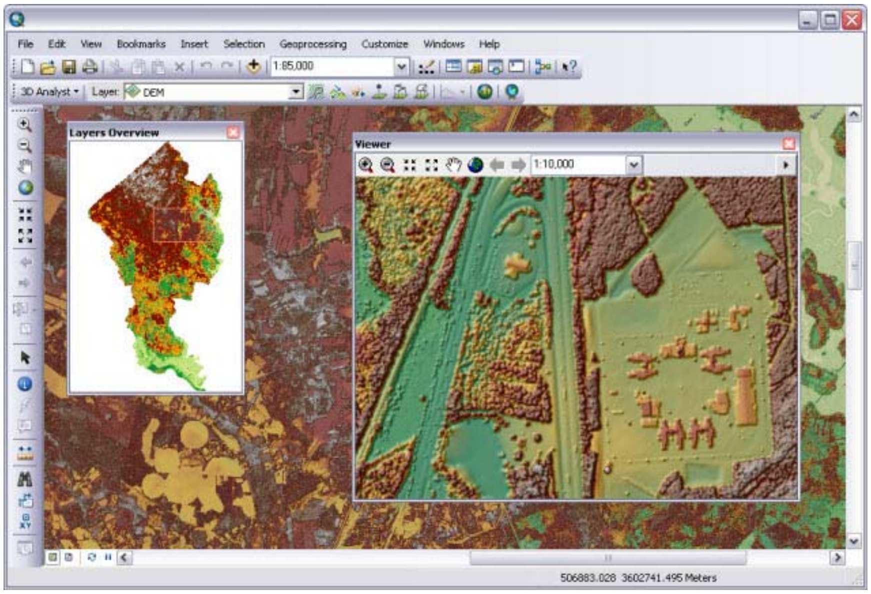
1. From the **3D Analyst Tools**, double-click the Terrain To Raster geoprocessing tool to open it.
2. Click the **Input Terrain** browse button to add the terrain dataset.
3. Click the **Output Raster** browse button to specify the location where the raster dataset is to be created.
4. Set the optional **Output data type** parameter to either 32-bit floating point or 32-bit integer. Floating point is the default value.
5. Set the interpolation method to either Linear or Natural Neighbors.
   These methods are TIN-based interpolation methods applied through the triangulated terrain surface. The Linear option finds the triangle encompassing each cell center and applies a weighted average of the triangle's nodes to interpolate a value. The Natural Neighbors option uses the Voronoi neighbors of cell centers.

   Consider the natural neighbors method for interpolating a terrain surface. Natural neighbor interpolation exhibits a longer processing time; however, the generated surface is much smoother than that produced with a linear interpolation. It is also less susceptible to small changes in the triangulation.
6. Set **Sampling Distance** to either **Observations** or **Cellsize**, which controls the horizontal resolution of the raster. Indicate the value next to the option once you have selected the desired method.
   The **Observations** method calculates the cell size for you based on the set value this number represents and the number of cells you want on the longest edge of the raster surface. You can set the cell size explicitly using the **Cellsize** option.
7. Set the resolution to use. The resolution parameter indicates which pyramid level of the terrain dataset to use for conversion. To output a raster dataset at full resolution, set this parameter to 0. Pyramid levels are defined using z-tolerance or window size, which represents the approximate resolution of the terrain dataset relative to the full resolution data.
8. Consider using the environment settings to explicitly control the extents of the DEM you want to generate. To extract a subset of the terrain, click the **Environments** button on the bottom of the geoprocessing tool. Click the **General Settings** tab and define the extent of

the output DEM.

Using either Point To Raster or Terrain To Raster geoprocessing tool, you can process hundreds of millions, even billions, of lidar points into high-resolution gridded DEMs and DSMs. These surface models can then be used with the large collection of raster tools available in ArcGIS for analysis.

They are also great for making maps (see graphic below) and, due to their simple data structure, are easy to share.


File
Edit
View
Bookmarks
Insert
Selection
Geoprocessing
Customize
Windows
Help
1:85,000
3D Analyst
Layer:
DEM
Layers Overview
Viewer
1:10,000
506883.028 3602741.495 Meters