What’s the tool for accrual?

Mosaic downloaded Lidar data OR Mosaic after DSM creation?

Solar insolation over Mosaiced DSM or smaller DSM and merge?

Tool for extracting insolation values from a raster with respect to road vector/Parking lot vector? Will the attribute value of the shape file get updated?

v.what.rast

Note: r.sun max value different in r.sun(DEM) and r.sun(DEM+deciduous). In fact max(r.sun(DEM)) < max(r.sun(DEM+deciduous)).

**Code**

Open GRASS 7.6 through OSGeo

pdal translate --input=16600\_1.las --writers.las.minor\_version=2 --output=16600\_1\_v12.las

Open normal GRASS GIS

# Coordinate system is 2264. Convert to 3358

lasinfo soccer\_field\_v\_12.las

las2las --a\_srs=EPSG:2264 --t\_srs=EPSG:3358 -i soccer\_field\_v\_12.las -o soccer\_field\_v\_12\_3358.las

# Removing noise from LAS file in lasnoise tool.

lasnoise -cpu64 -i "C:\Users\vishn\Google Drive\NCSU\_Courseware\GIS\_714\_Geocomputing and Simulations\Final\_project\Data\Study\_area\_Lidar\_data\study\_area\_lidar\_data\_v12\_3358.las" -step\_xy 4 -step\_z 1 -remove\_noise -odir "C:\Users\vishn\Google Drive\NCSU\_Courseware\GIS\_714\_Geocomputing and Simulations\Final\_project\Data\Study\_area\_Lidar\_data" -odix "\_denoised" -olas

#Computing DSM

r.in.lidar -e -n input=C:\Users\vishn\Google Drive\NCSU\_Courseware\GIS\_714\_Geocomputing and

Simulations\Final\_project\Data\Study\_area\_Lidar\_data\study\_area\_lidar\_data\_v12\_3358\_denoised.las output=dsm\_1m method=max resolution=1 zrange=0,275 –overwrite

# Gap filling

r.fill.stats -k input=dsm\_50cm@PERMANENT output=dsm\_50cm\_filled distance=3 power=2.0 cells=8

Ran the above code 6 times (varying inputs: dsm\_50cm \_filled1, dsm\_50cm \_filled2, dsm\_50cm \_filled3, dsm\_50cm \_filled4, dsm\_50cm \_filled5 ) to finally obtain dsm\_50cm\_filled6.

# r.sun

Mode 2

r.sun elevation=dsm\_50cm\_filled6@PERMANENT glob\_rad=jun21\_total\_irradiance\_50cm insol\_time=jun21\_insolation\_50cm day=172 nprocs=6

# importing parking lots shapefile into GRASS

v.in.ogr input=C:\Users\vishn\Google Drive\NCSU\_Courseware\GIS\_714\_Geocomputing and Simulations\Final\_project\Data\Parking\_Areas\Parking\_Areas\_3358\_clipped.shp output=parking\_lots

# Extracting values from raster

v.rast.stats map=parking\_lots@PERMANENT raster=jun21\_total\_irradiance@PERMANENT column\_prefix=Total\_jun21

This will create many unnecessary columns than necessary.

# changing color of vector file

v.colors map=parking\_lots@PERMANENT use=attr column=Total\_jun21\_sum color=bcyr

For outline of polygons only: double click on parking\_lots@PERMANENT, (d.vect pops up). Selection 🡪 layer =-1. Colors🡪 Area fill color 🡪 Transparent

# Exporting as a Geopackage instead of shapefile

v.out.ogr -c input=parking\_lots@PERMANENT output=C:\Users\vishn\Google Drive\NCSU\_Courseware\GIS\_714\_Geocomputing and Simulations\Final\_project\Data\Outputs\parking\_lots\_solar\_jun21.gpkg format=GPKG

r.in.lidar input=soccer\_field\_v\_12\_3358.las output=count\_1 method=max -e -n resolution=1

# Fill data gaps

r.fill.stats input=count\_1@PERMANENT output=count1\_filled distance=3 mode=wmean power=2.0 –overwrite

# Gaps not filled entirely. Repeat again on the newly filled raster

r.fill.stats input=count1\_filled@PERMANENT output=count1\_filled2@PERMANENT distance=3 mode=wmean power=2.0 cells=8

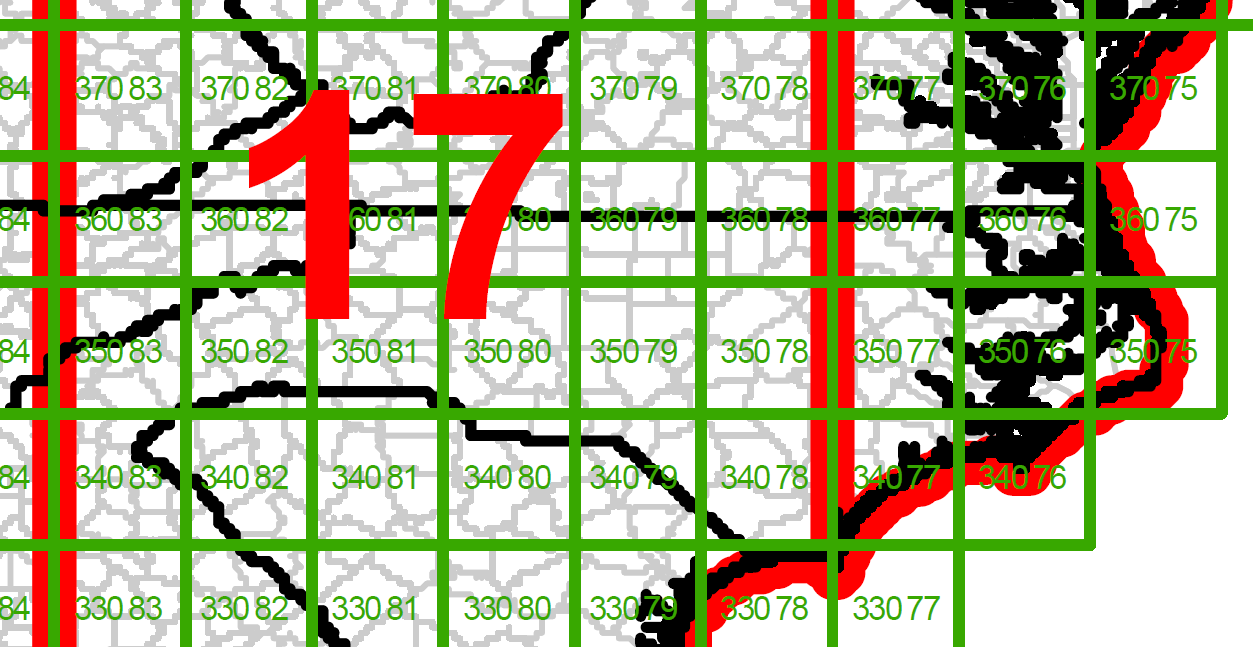
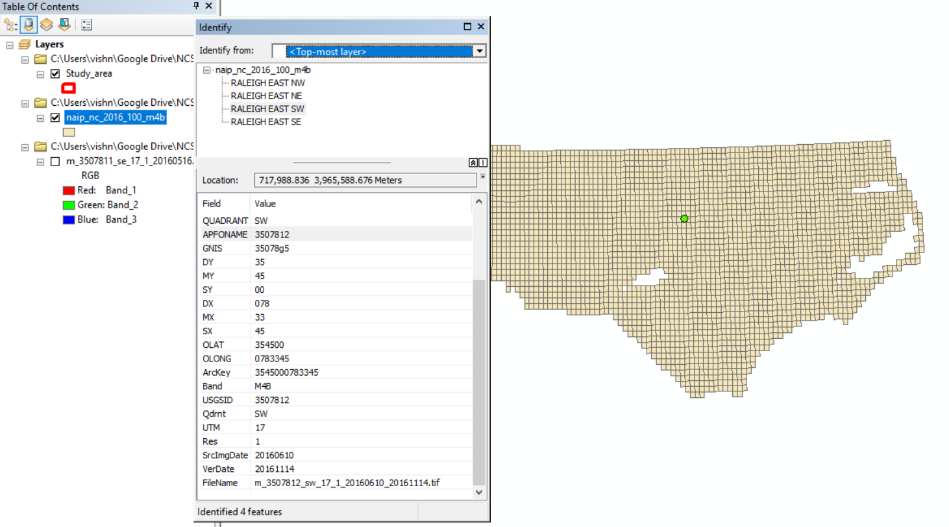
# r.fill and r.stats reducing quality of outputs.

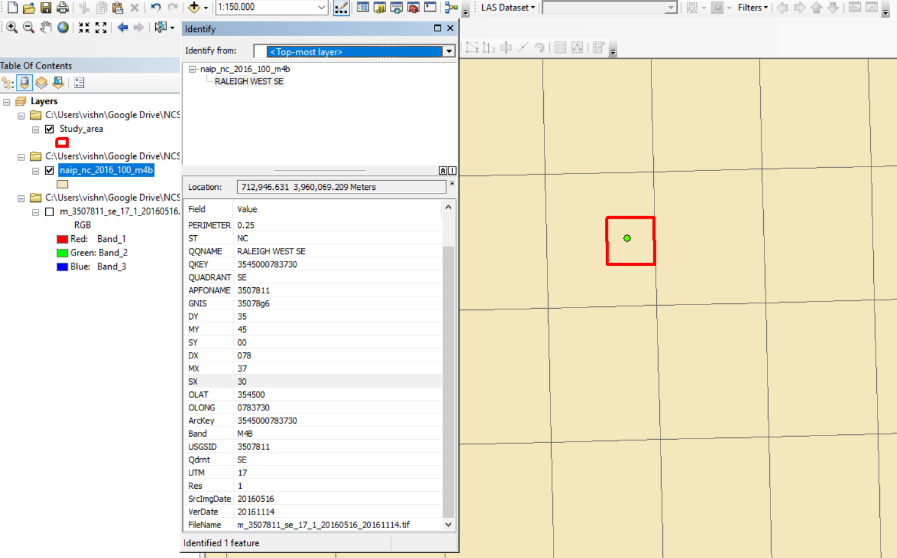
# Trying with v.in.lidar

25Mar19

# Vegetation only from Lidar

r.in.lidar input=C:\Users\vishn\Google Drive\NCSU\_Courseware\GIS\_714\_Geocomputing and Simulations\Final\_project\Data\Study\_area\_Lidar\_data\study\_area\_lidar\_data\_v12\_3358\_denoised.las output=Vegetation\_3\_4\_5 method=max class\_filter=3,4,5 resolution=1 –o

NAIP Imagery



To create a tree mask ( or a raster with trees only(tree heights only)), we could have used the “r.in.Lidar” with classification categories of vegetation only. But, it is very difficult to fill holes within the vegetation raster without growing the canopy circumference. No ready made algorithms exists in GRASS/ArcGIS/R. Tried the methods given here <https://stackoverflow.com/questions/41186218/fill-raster-holes-in-r-or-grass-gis>. But R has memory constraints, which means that the study area needs to be split into very small pieces, which won’t be good. So the approach to create a Tree Mask is shown below.

Fill zero values in “Trees eliminated r.sun\_output” with value 1216.

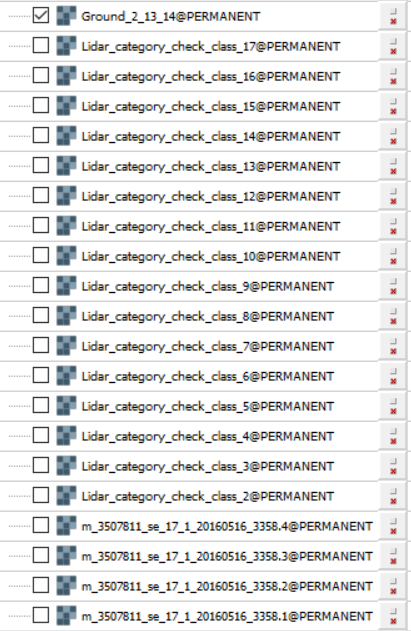
Trees eliminated r.sun\_output = r.sun\_output \* Compliment of Tree Mask

The Lidar points classification is bad in QL2 data as shown in table below.

Tree Mask = if (DSM- (DEM + Building heights) >2 ? 1 : 0

|  |  |  |
| --- | --- | --- |
| Classification value | Meaning | QL2 data from Spatial Data Download |
| 0 | Never classified |  |
| 1 | Unassigned |  |
| 2 | Ground | Ground |
| 3 | Low Vegetation | Low Vegetation |
| 4 | Medium Vegetation | Medium Vegetation |
| 5 | High Vegetation | High Vegetation |
| 6 | Building | Building |
| 7 | Low Point |  |
| 8 | Reserved  \* |  |
| 9 | Water |  |
| 10 | Rail |  |
| 11 | Road Surface |  |
| 12 | Reserved  \* |  |
| 13 | Wire - Guard (Shield) | Road Surface |
| 14 | Wire - Conductor (Phase) | Bridge Deck |
| 15 | Transmission Tower |  |
| 16 | Wire-Structure Connector (Insulator) |  |
| 17 | Bridge Deck | High Noise |
| 18 | High Noise |  |
| 19-63 | Reserved |  |
| 64-255 | User Definable |  |

Each Lidar classification was individually checked to see what class the classification value belong to. See picture below.



# Calculating the DEM

r.in.lidar --overwrite input=C:\Users\vishn\Google Drive\NCSU\_Courseware\GIS\_714\_Geocomputing and Simulations\Final\_project\Data\Study\_area\_Lidar\_data\study\_area\_lidar\_data\_v12\_3358\_denoised.las output=Ground\_2\_13\_14 method=min resolution=0.5 class\_filter=2,13,14

This was then smoothed many times using r.fill.stats. Ground\_2\_13\_14\_filled12 is the final filled DEM.(Please note that this file still have No data values in area belonged by buildings(some building area got filled while some didn’t). The unfilled building areas (with No Data values) were filled with a value 1. This could have been avoided if the r.fill.stats could have run around 100 times, which is cumbersome.)

r.mapcalc --overwrite expression=DSM\_minus\_DEM = dsm\_50cm\_filled6@PERMANENT - Ground\_2\_13\_14\_filled12@PERMANENT >= 2? dsm\_50cm\_filled6@PERMANENT - Ground\_2\_13\_14\_filled12@PERMANENT:0

r.mapcalc expression=Tree\_and\_portion\_of\_buildings=if(DSM\_minus\_DEM>2,1,0)

02Apr19

# Creating building mask

v.to.rast input=Raleigh\_Buildings\_clip@PERMANENT type=area output=Buildings\_mask use=val

r.null map=Buildings\_mask@PERMANENT null=0

# Creating a DEM with building heights added to it

r.mapcalc expression=dsm\_buildings\_only = dsm\_50cm\_filled6@PERMANENT \* Buildings\_mask@PERMANENT

r.mapcalc --overwrite expression=DEM\_with\_building\_heights = Ground\_2\_13\_14\_filled12+dsm\_buildings\_only

# Creating tree mask

r.mapcalc expression=Trees = (dsm\_50cm\_filled6@PERMANENT-DEM\_with\_building\_heights@PERMANENT)>2?1:0

# Creating inverted tree mask

# Removing tree-values from jun21\_total\_irradiance\_50cm by multiplying with mask

r.mapcalc expression=jun21\_total\_irradiance\_50cm\_trees\_removed = jun21\_total\_irradiance\_50cm@PERMANENT \* Trees\_inverted@PERMANENT

# Substituting a value of 1216 (the least value for a pixel under 24 hours of shade) for tree pixels

r.mapcalc expression=jun21\_total\_irradiance\_50cm\_trees\_adjusted = jun21\_total\_irradiance\_50cm\_trees\_removed@PERMANENT==0?1216:jun21\_total\_irradiance\_50cm\_trees\_removed

# Extracting values from raster

v.rast.stats map=Parking\_Areas\_3358\_clipped@PERMANENT raster=jun21\_total\_irradiance\_50cm\_trees\_adjusted@PERMANENT column\_prefix=Total\_irr\_Jun21\_ method=minimum,maximum,average,sum

NDVI from NAIP will not work out in our study because NAIP doesn’t seem to be orthorectified. So the tree mask from Lidar data may not exactly overlap with tree top from NAIP.

3 band orthorectified imagery (6 inch resolution) from NConemap (<http://data.nconemap.gov/geoportal/catalog/raster/download.page>) was used.

Greenness Index was used instead of NDVI.

Greenness Index = Green (Red + Green + Blue) (refer fig 3c from “Individual Urban Tree Species Classification Using Very High Spatial Resolution Airborne Multi-Spectral Imagery Using Longitudinal Profiles”)

Creating GI and problems faced

1) Raster calculator runs in ArcGIS, but no output. So QGIS was the next best choice. All the 6 rasters where exported to Tiff format from SID format in ArcGIS and then reprojected to 3358. (Choose the option “No” if ArcGIS asks to alter pixel depth)  
2) QGIS needs to be closed after running raster calculator once. In total 6 raster calculator operations were needed to compute GI for 6 high resolution rasters.  
3) No direct tools for mosaicking in ArcGIS. Raster to Mosaic didn’t work. Raster calculator can be used where the component rasters can be added together. Needs to set extent to the extent of all the 6 rasters.

Mosaicing in GRASS

# import all the 6 GI rasters into GRASS one by one

r.import input=C:\Users\vishn\Google Drive\NCSU\_Courseware\GIS\_714\_Geocomputing and Simulations\Final\_project\Data\6\_inch\_resolution\_Imagery\20170403\_3358\_GI.tif output=20170403\_3358\_GI

# set the computational region to an extent encompassing all the 4 rasters

g.region raster=20170301\_3358\_GI@PERMANENT,20170302\_3358\_GI@PERMANENT,20170403\_3358\_GI@PERMANENT,20170404\_3358\_GI@PERMANENT,20171301\_3358\_GI@PERMANENT,20171403\_3358\_GI@PERMANENT

#mosaicking using r.patch command

r.patch r.patch --overwrite input=20170301\_3358\_GI@PERMANENT,20170302\_3358\_GI@PERMANENT,20170403\_3358\_GI@PERMANENT,20170404\_3358\_GI@PERMANENT,20171301\_3358\_GI@PERMANENT,20171403\_3358\_GI@PERMANENT output=GI\_patched

# Set the computational region with extent = tree mask and resolution = 0.5

g.region raster=Trees@PERMANENT res=0.5

# Create GI raster of trees alone using tree mask.

r.mapcalc expression=GI\_patched\_tree\_mask = GI\_patched@PERMANENT \* Trees@PERMANENT

# Create GI raster of evergreen trees. 0.375 is the GI threshold value for evergreen trees

r.mapcalc expression=GI\_patched\_tree\_mask\_evergreen = GI\_patched\_tree\_mask@PERMANENT >=0.375

# Create GI raster of deciduous trees.

r.mapcalc --overwrite expression=GI\_patched\_tree\_mask\_deciduous = GI\_patched\_tree\_mask@PERMANENT <0.375 && GI\_patched\_tree\_mask@PERMANENT !=0

# Extract height of deciduous trees from DSM

r.mapcalc expression= heights\_of\_deciduous\_only = dsm\_50cm\_filled6@PERMANENT \* GI\_patched\_tree\_mask\_deciduous@PERMANENT

# DEM+deciduous

r.mapcalc expression=DEM\_plus\_deciduous = if(heights\_of\_deciduous\_only @PERMANENT>0, heights\_of\_deciduous\_only @PERMANENT, Ground\_2\_13\_14\_filled\_iteration50@PERMANENT)

# r.sun on DEM+deciduous

# Extract height of evergreen trees from DSM

r.mapcalc expression=heights\_of\_evergreen\_only = dsm\_50cm\_filled6@PERMANENT \* GI\_patched\_tree\_mask\_evergreen@PERMANENT

# DEM+evergreen

r.mapcalc expression=DEM\_plus\_evergreen = if( heights\_of\_evergreen\_only@PERMANENT >0, heights\_of\_evergreen\_only@PERMANENT, Ground\_2\_13\_14\_filled\_iteration50@PERMANENT)

# r.sun on DEM+evergreen

r.sun elevation=DEM\_plus\_evergreen@PERMANENT glob\_rad=Evergreen\_total\_irradiance\_01jan insol\_time=Evergreen\_insolation\_01jan day=1 nprocs=6

# creating a mask (or extracting) total irradiance of pixels that falls under deciduous trees at any time of the day. The rest of the pixels are assigned zero.

r.mapcalc expression=brightest\_pixels\_removed\_from\_deciduous\_shade = if( Deciduous\_total\_irradiance\_01jan@PERMANENT >= 2989 ,0,Deciduous\_total\_irradiance\_01jan@PERMANENT)

# Adjusting by a factor of 0.66

r.mapcalc expression= brightest\_pixels\_removed\_from\_deciduous\_shade\_adjusted\_by\_066 = brightest\_pixels\_removed\_from\_deciduous\_shade @PERMANENT + 0.66\*(3119- brightest\_pixels\_removed\_from\_deciduous\_shade @PERMANENT)

# Substituting zero values with r.sun(DEM) value. Zero values have become 2058.54 now.

r.mapcalc --overwrite expression=Adjusted\_total\_irradiance\_by\_dem\_plus\_deciduous\_01jan = if( brightest\_pixels\_removed\_from\_deciduous\_shade\_adjusted\_by\_066@PERMANENT ==2058.54, DEM\_total\_irradiance\_01jan@PERMANENT , brightest\_pixels\_removed\_from\_deciduous\_shade\_adjusted\_by\_066@PERMANENT )

## Taking compliment of evergreen Tree mask.

r.mapcalc expression=GI\_patched\_tree\_mask\_evergreen\_compliment = if(GI\_patched\_tree\_mask\_evergreen@PERMANENT==0,1,0 )

#Tree-pixel removal for evergreen

r.mapcalc expression=Evergreen\_total\_irradiance\_01jan\_zero\_beneath\_trees = Evergreen\_total\_irradiance\_01jan@PERMANENT\* GI\_patched\_tree\_mask\_evergreen\_compliment@PERMANENT

r.mapcalc expression=Evergreen\_total\_irradiance\_01jan\_zero\_beneath\_trees\_adjusted = if(Evergreen\_total\_irradiance\_01jan\_zero\_beneath\_trees@PERMANENT==0,678,Evergreen\_total\_irradiance\_01jan\_zero\_beneath\_trees)

## Taking compliment of evergreen Tree mask.

r.mapcalc expression=GI\_patched\_tree\_mask\_deciduous\_compliment = if(GI\_patched\_tree\_mask\_deciduous@PERMANENT==0,1,0)

#Tree-pixel removal for adjusted deciduous and substituting zero values with 2280 [New pixel value = min + 0.66 \*(max-min) ] Here min = 678 and max = 3105

r.mapcalc expression=Adjusted\_total\_irradiance\_by\_dem\_plus\_deciduous\_01jan\_zero\_beneath\_trees = Adjusted\_total\_irradiance\_by\_dem\_plus\_deciduous\_01jan@PERMANENT\* GI\_patched\_tree\_mask\_deciduous\_compliment@PERMANENT

r.mapcalc expression=Adjusted\_total\_irradiance\_by\_dem\_plus\_deciduous\_01jan\_zero\_beneath\_trees\_adjusted = if(Adjusted\_total\_irradiance\_by\_dem\_plus\_deciduous\_01jan\_zero\_beneath\_trees@PERMANENT==0,2280,Adjusted\_total\_irradiance\_by\_dem\_plus\_deciduous\_01jan\_zero\_beneath\_trees)

# Finding the minimum among the three

r.mapcalc --overwrite expression=Total\_irradiance\_minm\_among\_all\_three\_01jan = min( Adjusted\_total\_irradiance\_by\_dem\_plus\_deciduous\_01jan\_zero\_beneath\_trees\_adjusted@PERMANENT , Evergreen\_total\_irradiance\_01jan\_zero\_beneath\_trees\_adjusted@PERMANENT , DEM\_plus\_buildings\_total\_irradiance\_01jan@PERMANENT )

# Extracting values from raster

v.rast.stats map=Parking\_Areas\_3358\_clipped@PERMANENT raster=Total\_irradiance\_minm\_among\_all\_three\_01jan@PERMANENT column\_prefix=Jan method=minimum,maximum,average

# Exporting as Geopackage

v.out.ogr input=Parking\_Areas\_3358\_clipped@PERMANENT output=C:\Users\vvivekn\Google Drive\NCSU\_Courseware\GIS\_714\_Geocomputing and Smulations\Final\_project\Outputs\Parking\_Areas\_solar\_irradiance\_Jun\_and\_Jan.gpkg format=GPKG

# Extracting values from raster (for Roads)

v.rast.stats map=Roads\_without\_medians\_3358\_clipped@PERMANENT raster=Total\_irradiance\_minm\_among\_all\_three\_01jan@PERMANENT column\_prefix=Jan method=minimum,maximum,average