Complete Streets: Making Pasadena streets more *safe* and *accessible* using Geographic Information Systems (GIS)

Juan Cardenas, David Lwin, Jordan Partida, James Simpson

Masters of Science in Geographic Information Science (MSGISci)
Department of Geography, California State University, Long Beach

Introduction

Complete Streets are designed and operated to enable safe access for all users, including pedestrians, bicyclists, motorists and transit riders of all ages and abilities.

Complete Streets elements Include:

- Visible Crosswalks
- Bus Alighting
- Sidewalks
- Pedestrian Signals
- Bicycle Lanes
- Pavement Conditions
- Amount of Traffic Collisions

The City of Pasadena requested a tool using GIS to easily visualize and prioritize streets that are in need of more Complete Street Elements.

Methodology

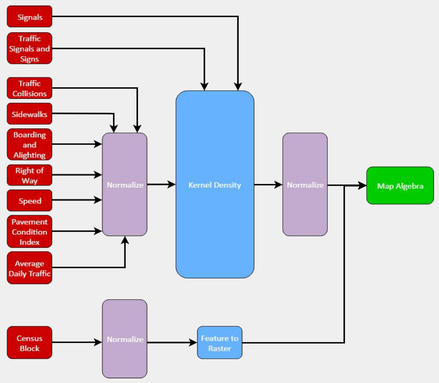

Figure 1. Methodology Flowchart representing the steps taken in the Complete Street Identification Tool using Python Scripting.

Data and Data Sources

Pasadena's scoring method was used as a reference for creating the normalized ranges used in the kernel density. All datasets were obtained from the City of Pasadena Public Data Portal.

Census Blocks (population count)	Range	0 - 861	862 - 1191	1192 - 1595	1596 - 2229	2330+
	Score	1	3	5	8	10
Right of Way (feet)	Range	0 - 15	16 - 30	31 - 45	46 - 70	71+
	Score	1	3	6	8	10
Speed (mph)	Range	0 - 25	26 - 30	31 - 35	36 - 40	41+
	Score	1	3	6	8	10
Pavement Condition Index	Range	0 - 39	40 - 57	58 - 74	75 - 85	86 - 100
	Score	1	2	5	8	10
Average Daily Traffic (cars per day (k))	Range	18k+	12 - 17.9k	9 - 11.9k	0 - 8.9k	
	Score	1	3	5	10	
Sidewalks	Range	YES	NO			
	Score	5	0			
Boarding & Alighting	Range	YES	NO			
	Score	7	0			
Signals	Range	YES	NO			
	Score	5	0			
Traffic Collisions	Range	*Normalized After Kernel Density				
	Score					
Traffic Signs	Range	*Normalized After Kernel Density				
	Score					

Table 1. Scoring table showing the different feature classes and data value ranges with their corresponding score value. Each score was based upon importance in Complete Street Design.

Results

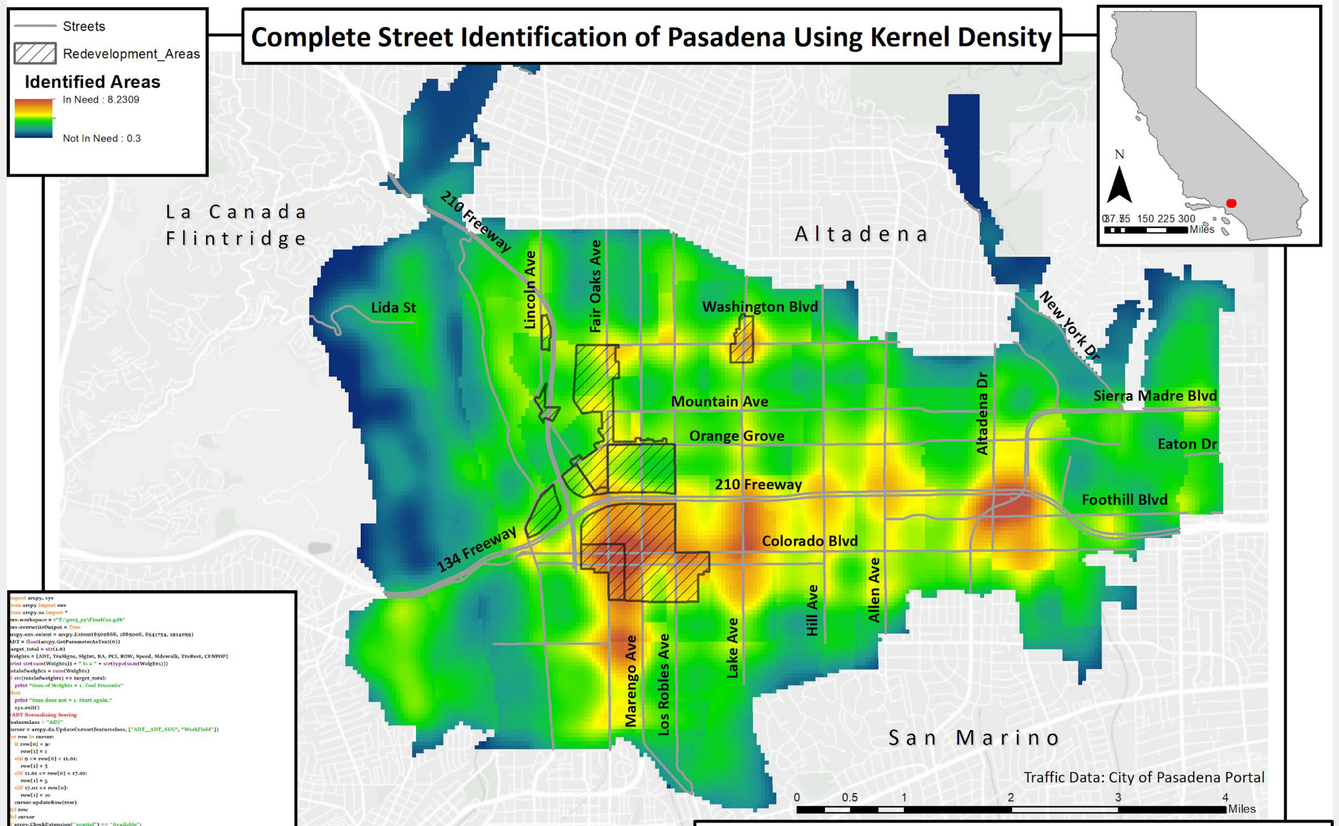

Figure 2. Weighted ouput of Kernel Density Rasters based off the City of Pasadena helps to identify areas in need of more Complete Street elements. Elements used in the Identification are; Average Daily Traffic,Traffic Collisions,Pavement Condition Index, Bus Boarding and Alighting, Speed, Signalized Intersections, Other Traffic Signals, Right of Way, and Sidewalks. The data is normalized on a scale of 0 - 10. Low values indicated in blue and green display streets with satisfying amounts of Complete Street elements. Higher values in red correlate to areas lacking Complete Street elements. This map output contains the default weight of ten percent. Results will vary depending on the user's input.

```
import arcpy, sys
from arcpy import env
from arcpy.sa import *
env.workspace = r"F:\jpm\_pyFinalCut.gdb"
env.overwriteOutput = True
arcpy.env.extent = arcpy.Extent(6502668, 1865006, 6541754, 1914059)
ADT = float(arcpy.GetParameterAsText(0))
target_total = str(1.0)
Weights = [ADT, TraSigns, SigInt, BA, PCI, ROW, Speed, Sidewalk, TraRest, CENPOP]
print str(sum(Weights)) + " is a " + str(type(sum(Weights)))
totalweights = sum(Weights)
if str(totalweights) == target_total:
  print "Sum of Weights = 1. Tool Proceeds"
else:
  print "Sum does not = 1. Start again."
  sys.exit()
# ADT Normalizing Scoring
featureclass = "ADT"
cursor = arcpy.da.UpdateCursor(featureclass, ["ADT__ADT_AVG", "WorkField"])
for row in cursor:
  if row[0] < 9:
    row[1] = 1
  elif 9 <= row[0] < 11.01:
    row[1] = 3
  elif 11.01 <= row[0] < 17.01:
    row[1] = 5
  elif 17.01 <= row[0]:
    row[1] = 10
  cursor.updateRow(row)
del row
del cursor
if arcpy.CheckExtension("spatial") == "Available":
  arcpy.CheckOutExtension("spatial")
# ADT Normalizing KD
in_features = "ADT"
out_rasterdataset = in_features + "Rast"
outKD_ADT = KernelDensity(in_features, "WorkField")
outKD_ADT.save(out_rasterdataset)
outKD_ADT_max = float(arcpy.GetRasterProperties_management(outKD_ADT, "MAXIMUM").getOutput(0))
print type(outKD_ADT_max)
outKD_ADT_div = outKD_ADT / outKD_ADT_max
Norm_ADT = outKD_ADT_div * 10
Norm_ADT.save("Norm_ADT")
# Map Algebra Equation for all feature classes
final_raster = (ADT * Norm_ADT)+(TraSigns * Norm_TC_signs)+(SigInt * Norm_Signals)+(BA * Norm_BA)
+(PCI * Norm_PCI)+(ROW * Norm_ROW)+(Speed * Norm_Speed)+(Sidewalk * Norm_Sidewalk)
+(TraRest * Norm_TC_rest)+(CENPOP * Census_Raster)
final_raster.save("final_output")
arcpy.SetParameterAsText(16, final_raster)
arcpy.CheckInExtension("spatial")
else:
  print "Spatial Analyst Extension is not available."
```

Figure 3. Complete Streets Identification sample script for Average Daily Traffic.

Discussion

A variety of methods were explored in order to develop a tool with results that identify areas in need of improvement.

Originally, Raster Weighted Overlay method was attempted but the results were limited and undesirable due to the inconsistencies in data.

Using the Kernel Density tool gave us more desirable results that are aesthetically pleasing and useful. The tool was developed using python coding.

The street centerlines for the feature classes in some areas were about 40 feet off. Better accuracy of data would have made a more accurate product.

Conclusion

The tool is useful in visualizing and prioritizing streets that are in need of improvement.
One of the limitations with the tool is that it does not show exactly which Complete Street elements are missing. The final product is based off the user's manual input of desired weights and is represented visually but not within the attribute table.

Some of the necessary feature classes were unable to be quantified in a way they could be used for raster processing.

The tool is not a transferable product in terms of universal use due to time constraints. For improvement, a web application can be developed so that the tool can be more accessible and user friendly.

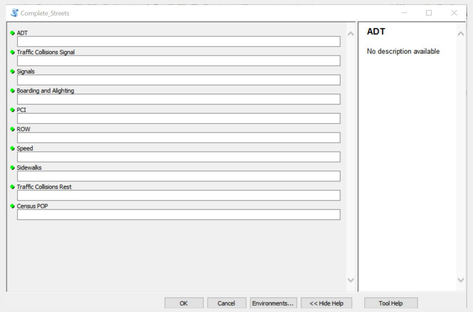

Figure 4. Graphical user interface for the user's weight input. This interface was written into the Python Complete Streets Identification Tool.

This project was requested by the southern california association of governments (SCAG) data science federation. Submitted in partial fulfillment of the requirements of the Masters of Science in Geographic Information Science(MSGISci), August 9, 2019.

For additional information please contact:
Juan Cardenas Jcardenas743@gmail.com
David Lwin Davod.Kyaw6@gmail.com
Jordan Partida Partida.jordan@gmail.com
James Simpson James.w.simpson25@gmail.com